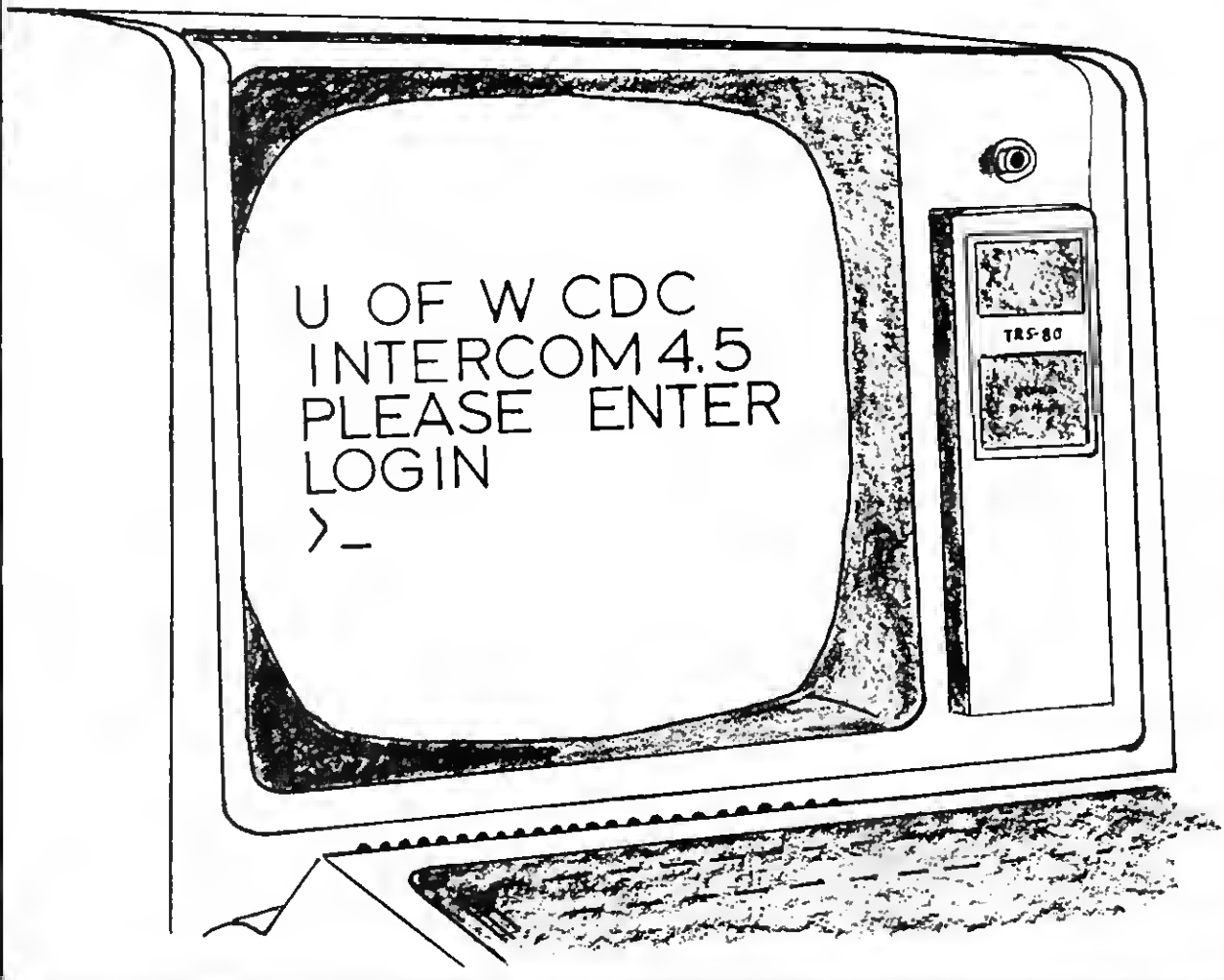
80-US

$3.00 PER COPY

The TRS-80 Users Journal

Vol. II, No. 2 MAR-APR 1979

IS THERE A MODEM IN YOUR FUTURE? ..... Page 11



ADVERTISERS-80-US will accept limited relevant commercial advertising that pertains to or for use with the TRS-80 Micro-computer. Please write for rate card.

WRITERS/CONTRIBUTORS

We are seeking material from writers. Send us your TRS-80 related material (except that which has been previously published). Any material, articles, programs, pertinent short stories, will be appreciated. If you think you cannot write, send them; we will edit and give you the credit. You may send programs on cassette (Level II preferred) or on disk - they will be returned if you provide return postage. Generous compensation will be made for well documented and non-trivial works that are accepted for publication.

Editor/Publisher ..... I. Mike Schmidt
Technical Editor ..... T. Rosenbaum
Associate Editor ..... T.R. Dettmann
Software Consultant ..... K. Schmidt
Software Development..... L. Christopherson
Editorial Assistant ..... K. Huston
Production Advisor..... R. Mulkey
Resources-In-Learning Consultant
G.B. Livingston, PhD.

Contributors:
R. Groth
J. Miller
G. Blank
R. Thompson

The 80-US Journal is published bi-monthly by *80-NW Publishing, 3110 North 31st Street, Tacoma, Washington 98407, telephone (206) 759-9642*

Subscription Rates:
U.S. and Canada - $16.00 per year
Foreign - $24.00 per year (sent Airmail)
Foreign subscriptions must be remitted in U.S. funds.
Printed in the United States of America.

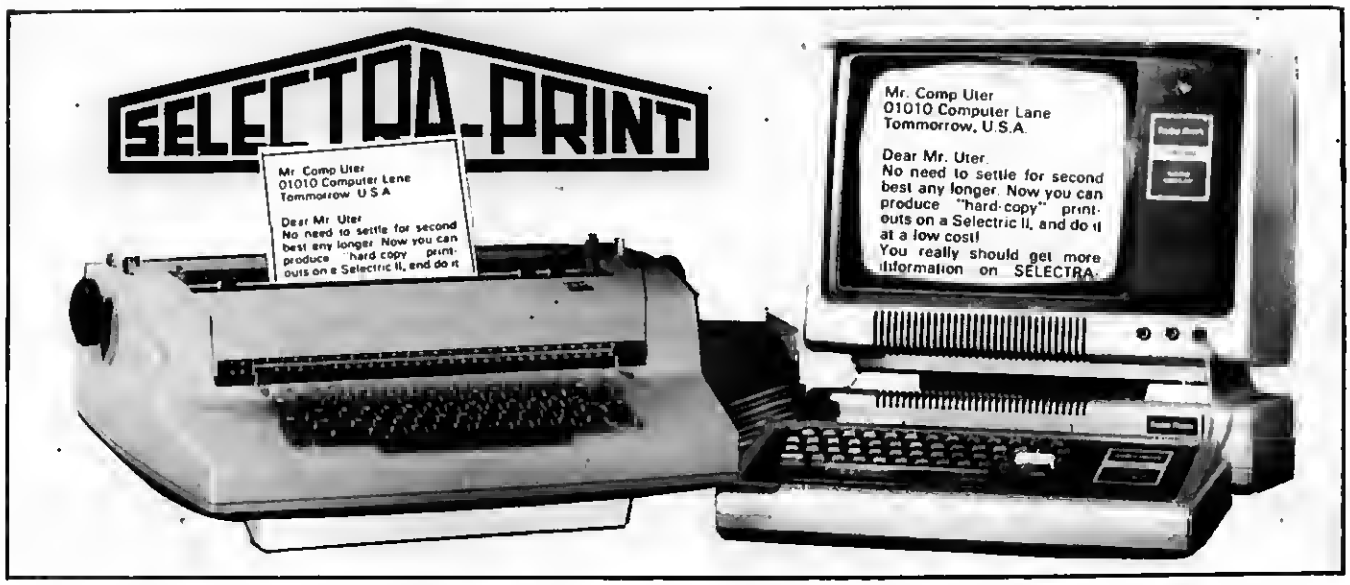
CLUBS, USER GROUPS & ORGANIZATIONS

We are offering special subscription rates to Clubs, User groups and institutions. Minimum participation number is ten. Write (on you club/organization letterhead please), for details concerning this special discount.

DEALERS

The JOURNAL is now a fully committed, international publication. You can do no better than to display this JOURNAL on your magazine rack. TRS-80 users/owners represent a very sizeable chunk of the business/personal computer market, and their numbers are growing at an astounding rate. Write or call for special Dealer discount information.

Please address all correspondence inquiries to:
80-US Journal
P.O. Box 7112
Tacoma, WA 98407



IF YOU OWN A TRS 80 *.....

TRS-80 is a product of Radio Shack, A Tandy Corporation

SELECTRA-PRINT PUTS YOU JUST A CABLE LENGTH AWAY FROM A "HARD—COPY" (Typewriter Quality: clean, clear, high-fidelity) Word Processing System!

. . .just a simple hook-up with the supplied cables and your *SELECTRA-PRINT* is ready to run.

For Word Processing on a TRS-80, for example, just command "LPRINT" and *SELECTRA-PRINT* automatically outputs clear, clean, high-fidelity, hard copy. . . . and of course you can use it to print-out any other information you might need.

SPECIAL NOTE:
SELECTRA-PRINT is a Selectric II typewriter and although it has been modified for computer print-out, it may still be used as a standard office typewriter.

PRICE: $1,850.00* *TRS-80 Version $1925.00

OPTIONS:

Dual Pitch	$125.00
Correction Feature	$125.00
Noise Reduction Feature	$50.00
RS-232 Interface	$195.00

* *SELECTRA-PRINT is VERSATILE!*
SELECTRA-PRINT is compatible with most micro- and mini-computers including *PET* ● *Apple* ● *Heath H8* ● *IMSAI* ● *Cromemco* ● *Alpha Microsystems* ● *Space Byte* ● *North Star Horizon* ● *SWTP* ● *Vector Graphic* ● *Sol* ● *Polymorphic* ● *Digital Group* ● *Ohio Scientific* ● *Altair* ● *Sorcerer Xitan* ● *Rex* ● *KIM* ● *EXORcisor* . . . .

"From now on, PRINT-OUT takes on a new meaning"

M30

micro computer devices inc.

960 E. Orangethorpe, Bldg. F
Anaheim, California 92801
Telephone (714) 992-2270

"Innovators to the Microcomputer Industry"

Direct International Sales:
International Sales Division
17648 Orna Drive
Granada Hills, CA 91344, USA

Discounts Available to
EDUCATIONAL ACCOUNTS
Contact: Delores Sun P.O. Box 8394
Ann Arbor, MI 48105 (313) 665-8514

BEYOND TRS-80™

When Microsoft put Level II BASIC on TRS-80, you got a glimpse of its full potential.

Now Microsoft introduces:

TRS-80 Fortran

and TRS-80 will never be the same!

Plus

TRS-80 FORTRAN includes the finest Z-80 development software available:

Z-80 Macro Assembler, versatile Text Editor, and Linking Loader.

Total price: Only $350.00

TO: Microsoft, 10800 NE Eighth, Suite 819, Bellevue, WA 98004

☐ **Send me free TRS-80 FORTRAN overview.**

☐ **Send me TRS-80 FORTRAN and Z-80 development software for $350.00.**

☐ Check enclosed ☐ Master Charge ☐ VISA

Card Number _____ Exp. Date _____

Cardholder's Signature _____

Name _____

Address _____

City _____ State _____ Zip _____

Clip the coupon and ORDER NOW, or send for free overview for more details about TRS-80 FORTRAN.

TRS-80 FORTRAN is supplied on two minidiskettes and requires a 32K system with one disk drive. Dealer inquiries invited.

MICROSOFT

10800 NE Eighth, Suite 819
Bellevue, WA 98004
DEPT.80

Editorial Remarks * *

GET RICH QUICK SOFTWARE SCHEMES

You probably will agree that telling about an experience and having an experience are two different things. However, we feel a certain obligation to tell it like it is concerning software sales.

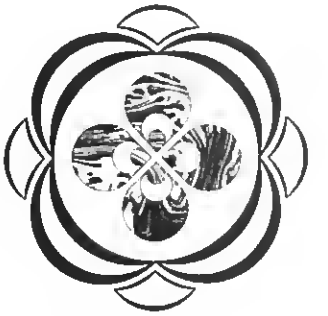
First off, the thousands of computers now being used have created a definite need for software of all kinds. We all bought the "monster", for various reasons, and now it needs to be fed. This makes anyone who can write even a small program a potential "supplier" of software, and looking at the various ads in print bears this out.

So you have written what you thinks is "the" program to end all programs - now what? You can give a copy to a friend , who will give a copy to a friend etc., and after a week, it will be all over the country, just like an off color joke. Or, you can send it off, have a master copy made (at up to $50.00 mastering fee), then have copies made (at up to $1.50 per copy), have labels printed (at $50.00 per thousand) and put them on by hand. Now you still have not sold one copy, because no one knows you have anything. Enter the wonderful world of advertising, tell the world you have a fantastic, super program for sale, right? That will only cost you $5.00 for an ad (which hardly anyone will see), or you can spend $400.00 plus for a quarter page in one of the "biggies" and get more response. But now you have to sell a bunch, just to break even, and Oops, what about something to mail those cassettes in - and the postage? Are you sure that program was as big a deal as you originally thought?

If you don't want to go this route you can always try to market your program through one of the large software concerns. We tried two of them, and without mentioning names, here is what happened: They want to have exclusive rights, just for openers. Then comes the agreement or contract. The agreement can be summarized pretty much like this: You assign, transfer, give and otherwise hand over, total and complete rights, copyrights and all other rights. In return for this you are assured of 5 or 10% of an undefined amount. (Isn't that just great? 20 percent of zero is zero!) And...you may find your program on a cassette with two or three other authors, in which case you have to split that percentage with them. Then to really cinch the deal, one agreement which we were offered had absolutely no mention of term of agreement, which means they may sit on your program forever, never publish it. When they do, have the right to keep it forever.

This was our first experience with two of the largest, and there is no way we can condemn all by these two. But there it is. Be careful, you may just be "had", and we think you ought to know about it. ...80

RANDOM ACCESS



Have you looked yet? It really is 48 pages this time. The comments concerning our "January White Sale" in the Jan-Feb issue were interesting, to say the least. Those eight white pages at the end were an accomplished fact before we knew about it, and there was no time to redo the run. The insert was the only graceful way out. And wouldn't you know it? Some readers suggested doing that all the time, for notes!

We welcome three new staff members to our pages this issue. PHIL PILGRIM is doing the SYSTEM/COMMAND column, and starts off with a very nice variable intensity program. TERRY DETTMANN, who was a contributor in the last issue, is now on the masthead, and presents two feature aricles this issue: IS THERE A MODEM IN YOUR FUTURE?, and FORTRAN. We will be seeing more from TERRY, and sometimes wonder when, if ever, he sleeps! Another regular will be GEORGE BLANK. GEORGE will primarily be doing software reviews, but he is also a prolific writer, and will be presenting aritcles and programs. Welcome aboard!

The May-Jun issue should be mailed as Second Class Mail. Getting things set up so that the printer automatically addresses and delivers to the post office is an extra we are working on. This should reduce the time in transit, so you should have your copies quite a bit sooner.

Some very interesting User Groups are, or have been, formed around the country. We will make a point of including some information on these as it becomes available (send in your club bulletin or newsletter!). One such is the MONTEREY BAY USERS GROUP, Pacific Grove, California. They publish a newsletter entitled TRS-80 MBUG Newsletter. First issue was Jan 79 and was four pages. Contacty Bill Pitt, PO Box GH, Pacific Grove, CA 93950 or (408) 373-7177 most evenings.

CHICATRUG, Chicago TRS-80 Users Group Newsletter, is a very nice publication with news, short programs and helpful tips and such. Manny Garcia is editor, and in his Jan issue he placed a quote which I liked so well I'll repeat it: On a clear disk, you can seek forever! How true. Contact Manny at 3950 N Lake Shore Drive Apt 2310, Chicago, Il 60612 or (312) 348-6562.

OCTUG, Orange County TRS-80 Users Group, puts out a very good newsletter, 24 pages in January, with plenty of good to know information and some program listings. Attendence at their last meeting was over 70! Contact Bill Barden, OCTUG, 24232 Tahoe Court, Laguna Niguel, CA 92677 or (714) 831-7004.

INSIDERS is a TRS-80 newsletter edited by Ray Daly in Washington DC which is heavy in machine language software plus aritcles of general interest. Contact Ray at 2617 42nd St NW #2, Washington, DC 20007 or (202) 337-4691.

In the NEW LITERATURE department, there is a flyer from ELECTRONIC SPECIALISTS which describes protection for your TRS-80. Protection against AC power line surges caused by large equipment or lightning is described. Also covered are causes and cures of AC line hash. Flyer TRS-PC is available for a stamped, self-addressed envelope. ELECTRONIC SPECIALISTS, Box 122, Natick, MA 01760.

ROBERT ELLIOTT PURSER's Reference List of TRS-80, PET & APPLE II Computer Cassettes should be in print by the time you read this. It is a list of over 2000 cassettes for sale or trade for the three computers mentioned. The Feb 79 issue sells for $4.00 (published quarterly). Robert Elliott Purser, PO Box 466, El Dorado, CA 95623.

The JOURNAL has been a full time occupation since the 1st of January 79. The phone is manned most days during normal working hours (Pacific Time) (206) 759-9642. If you can't get an answer, try again, as we are often busy with the printer, the Post Office, etc.

Don't forget to tell them you saw it in THE 80-US JOURNAL!

LETTERS

Gentlemen,
I have not as yet purchased a TRS-80 because i wanted to take a long hard look at what the market had to offer. Your publication is just the ticket for me to observe how this machine is being used, by you and others, as well as what new non-Radio Shack peripherals are being offered for the TRS-80.
Your purchase of a Selectra-Print was of particular interest to me. My main use for a micro-computer will be for accounting and financial report preparation. For this I will need clean clear copies. I had not considered using a converted Selectra because of what i read in 'So You Want to Buy A Computer", by R W Brown. In this book the author says:
"Stay away from the converted selectric typewriters. These converted machines do not work very reliably with most computer systems and break down frequently. IBM does not recommend their machines for this type of use and even if you can get a service contract they will not work on any part of the machine that has been modified. If you are unlucky you are apt to find your Basic or Fortran language will not support the driver program. Another problem is that the keyboard on the typewriter rarely works so it will not talk to your computer, so do not count on it to substitute as a single full terminal. To top it off why do you think these typewriters are flooding the market? IBM has stopped using the heavy duty models on their systems because they were down too often."
After you have used your Selectra-Print for a few months, how about informing your readers about its performance?
Best of luck with your publication.

Joseph M Helzmann
Rochester, NY

(Speeking only for the Selectre-Print, it will support eny lenguege you went to output to it. The Selectric typewriter portion of the system is under werrenty for 90 deys efter you receive it. We heve hed ours since ebout the middle of November 78, and thus fer heve hed to cell for service twice; once for e line feed problem end once for timing edjustment (it wes printing underscores end deshes occesionelly). Both were covered under the werrenty, which will heve expired by the time you reed this. For our use, we cen't think of eny other mechine which will give the quelity we went et the price.
Ed.)

Editor,
I was quite interested in the article on page 8 of the Nov-Dec 78 issue entitled "Talk to your Printer". I had recently purchased the TRS232 printer driver from Small Systems Sottware and was anxious to acquire printouts of the various programs thet i have on disk. Unfortunately the "route" command in the TRSDOS does not function - and I was unable to route directly information to the printer until I read your article.
I am also happy to report to those who might be interested in purchasing the TRS232, that i was successful in hooking an ASC33 teletype to my TRS-80. Although the ASC33 is extremely noisy, at least I have hardcopy of my basic programs and where needed, directory listings.

Raymond H. Colton
Syracuse, NY

Sir,
This is only my second copy of your fine magazine and i think it is the best magazine on computers I have seen yet, keep up the good work.

Bobby G Corder
Clovis, NM

Mike,
I enjoyed your "January White Sale". in an ancient Ugaritic clay tablet (source not verified), it stands written: "When they run you out of town on a rail, act like you are leading the parade." Looks like you have studied the ancient wisdom.

George Blank
Leechburg, PA

(Just couldn't resist getting your quote into print, but i like the quote better then the reeson for your quoting it! Ed.)

Mike,
I just received the Jan-Feb issue of 80-US and think you have done a fantastic job, both in the choice and the presentation ot the material. You also seem to be attracting a high caliber list of advertisers. i have just sent out a (bundle of checks) to them.

Michael F Drury
Fairbanks, AK

The reason M Schmidt's method for "renewing" programs (80-US Jan-Feb 79) is not always successful is as follows: There are not one but 4 pointers which must be altered.
1) 16633,4 40F9,A End of program pointer
2) 16635,6 40FB,C End of simple variables
3) 16637,8 40FD,E End of erray variables
4) 17129,30 42E9,A Level II pointer to 2nd line 26810,11 68BA,B Disk basic pointer to 2nd line of progrem.
Here is a procedure which works in L2 (even on ANDROID NIM!)
1) With program in memory, type CLEAR:?PEEK (16633)PEEK(16634), PEEK(17129)PEEK(17130)
1) Write down the four numbers obteined in 1)
3) Run the program (optional)
4) (Here it comes!) type NEW(enter)
5) Enter the following: POKE 16333 A:POKE 16634, B:POKE17129,C:POKE17130,D (where ABC&D are not letters, but the numbers you wrote down in step 2 above. Now type CLEAR (enter).
6) LIST & RUN the renewed program.
The CLEAR command in step 5 adjusts 40FB and 40FD as required. Caution: Between step 4 and step 5, do not use any commands which refer to variables. The reason is simple, after typing NEW, the end of the program pointer is changed to 42EB (68BC for DOS). Thus all variables referenced will be stored starting at this location - right on top of your old program!

James Garon
Fullerton, CA

(Very good, end thenks! Also of interest is the following, by Jemes in the Feb 79 issue of the Orenge County TRS-80 Users Group newsletter: "The word THEN cen elweys be repleced by e comme es in 200 IF X = 7,350 No, thet's not the number 7350 smuggling illegel punctuetion ecross the stete line, it's our new abbreviation for THEN. If X is seven, the progrem will brench to 350" For informetion regerding the OCTUG newsletter, contect them et 24232 Tehoe Court, Leguna Niguel Celifornie 92677
Ed.)

That Mailing List program (Nov-Dec 78 80-US) was dynamite! Worth the whole subscription price.

Dr. Allen Blumenthal
Plainfield, NJ

(Continued on page 8)

(In our very first issue, we promised to give our readers something of value equal to the price of a subscription in each issue - and we will keep right on with that philosophy. Ed.)

Daar Sir,
Tha Android Nim tapa arrived yesterday and is defactiva. I had expected to enjoy a scene of Androidai Massacre but after loading side A and initiating the RUN I soon ran into an UL arror in line 9. Side B would not load.

Newton M Gray
Seattla, Wa.

(We have found so far, that only 6 out of about 1000 tapes were bad. At least, that is the number we have had returned. Hopefully, anyone who has a bad one will raturn it, as we will replace them immediately, as was Mr. Gray's. We wonder how much of this happens in transit. Ed.)

Daar Sirs,
As a new subscriber, I have just racaivad my first issua of your magazine and anjoyad it very much. I have one question - is it actually possible to beat that Android Nim game?

WO Eden
Eugana, Or

(Winning at Android Nim seems to be the question from many of our raaders. Actually, it is, but you have to know the winning strategy and apply it aarly in the game. For now, try to leave the computer with an even number in two rows, then duplicate its moves. Or, try to leava it with one in one row, two in another and three in the other, which will normally get you to an even number in two rows. In the May-Jun issue, we will carry an article entitled "Chinese, Android and other Nims", which should give you all the revenge you seek. Ed.)

Hi,
The decision was mada, and our office went ahead with buying a TRS-80. I think your Journal encouraged that choica.
We have one loan written with interest to be computed on a 365 day year. It needs to consider the varying days in each month. More and more, we hear talk about interest based on a 365 day year, as loans rise with increasing property values. Do you have any programs for amortizing a loan with monthly principal and interast payments on the basis of 365 days?

LW Hardy
Los Gatos, CA

(Help anyone? You can correspond directly with Ms. Hardy at 15700 Gardenia Way, Los Gatos, CA. 95030. Ed.)

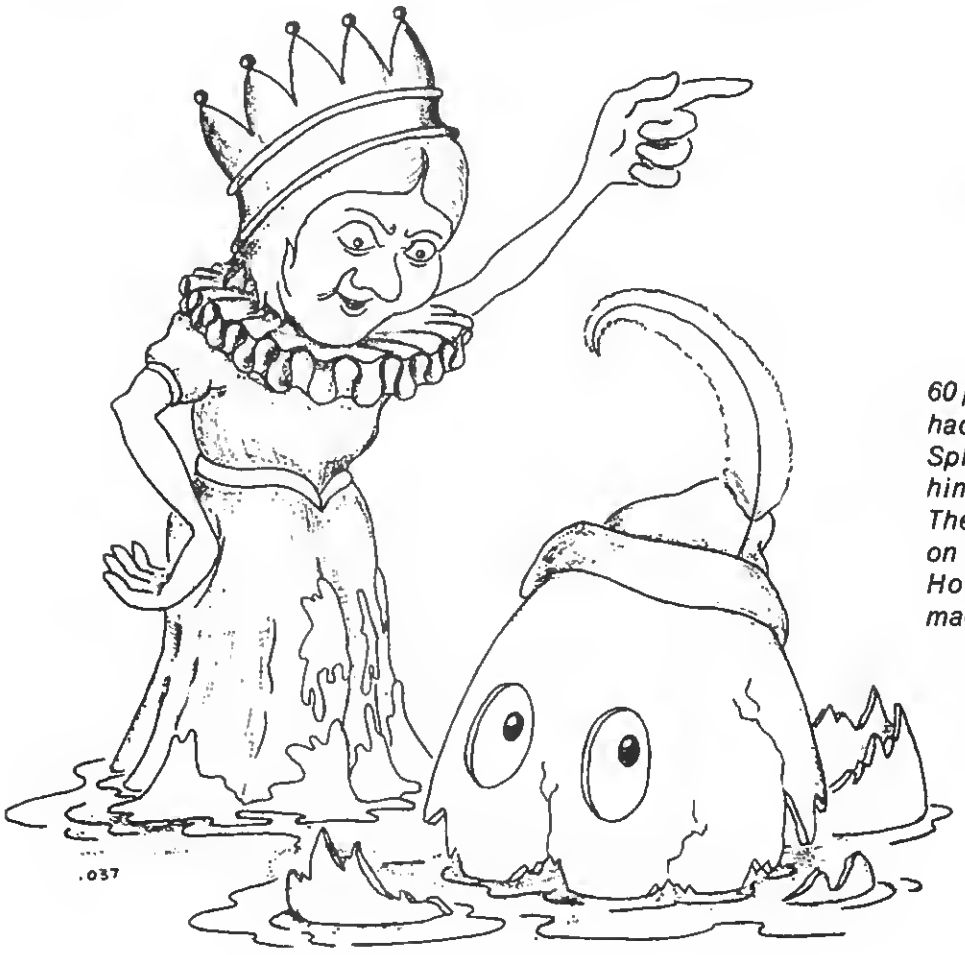
Mike,
I thoroughly enjoyed the Nov-Dec issue and am looking forward to the machine Language tuturial as I have been struggling for some time to learn machine language on my own, and it is tough! I'm busy writing a dis-assembler so that I can take apart L2 ROM in order to find tha keyboard, cursor, etc. routines so that I won't have to POKE my program in with a loop reading DATA statements. Tha new TI "Programmer" calculator makes this barely possible by easily doing all tha Hex to Decimal conversions. Kaap up the good work.

Louise H Frankenberg
Pasadena, MD

Gentlaman,
Thank you for the prompt service. I received my first copy of 80-US and copy of Android Nim yesterday. In all fairness, you should know that all publishers always make subscribers wait 6 to 8 weeks before they send an issua!!

Marvin J Krause
Sitka, AK

(We try harder and all that - but we are slipping, the current wait is about 10 days Ed.)



FOR THE LITTLE NYBBLERS

60 pound Humpty D.
had a great fall...
Splattered 90% of
himself over all
The surroundings and
on the Queen's dress.
How much of Humpty D.
made this great mess?

L.B. Christopherson

The 80-US JOURNAL

VOL. II, NO. 2 March-April 1979

Consecutive Issue Number Four

CONTENTS

TRS-80 HARDWARE & SOFTWARE

MEMORY 16K memory kit complete with 8 RAM chips, jumpers and instructions for installation in your keyboard or expansion interface. A household screwdriver is the only tool you need, no soldering... **$85.00**

PRINTERS

CENTRONICS: Nationwide field repair provided by Centronics. Call Centronics: 800/258-1952.

779 friction feed - 60cps - same as line printer used by Radio Shack... **$999.00**

779 tractor feed - recommended for business... **$1179.00**

700 tractor feed - 60 cps unidirectional 132 columns. **$1379.00**

701 tractor feed - 60 cps bidirectional 132 columns... **$1579.00**

Pf micro printer - same as Quick Printer used by Radio Shack... **$399.00**

Cable to plug into expansion interface... **$39.00**

Integraf Data Systems: 100 cps unidirectional - upper & lower case standard - switch selectable RS232 or Parallel - maximum paper with 8.5" edge to-edge - can attach to expansion interface or to keyboard with TRS 232 printer interface. (below)
IP125 - friction feed with variable pitch under program control 64 - 132 clms. per line. **$836.00**

IP225 - tractor feed with variable pitch, 2K buffer & full graphics options. **$1148.00**

Cable to plug into expansion interface. **$49.00**

Cable to plug into TRS232 interface. **$15.00**

TRS232 PRINTER INTERFACE (by Small System Hardware) Software driven serial output port plugs into keyboard cassette port (cassette plugs into TRS232), RS232 or current loop - can be used to drive Integral Data & Diablo printers, teletypes, etc. - also works with Electric Pencil... **$49.95**

DISK DRIVES AND DISKETTES

Shugart SA400 mini disk drives (same as used by Radio Shack) complete with power supply, case, & plug in instructions... **$399.00**

Disk drive cable for 4 drives (buy 2 drives & get cable free) $45.00

Diskettes - Verbatim... $3.60

Plastic case - holds 10 diskettes - built in flip open stand for easy access... $5.00

ELECTRIC PENCIL by Michael Shrayer - machine language cassette based word processor - includes instructions for modifying keyboard to set lower/upper case - can be used with TRS232 - by far the best word processing program we've seen!... **$99.95**

MICROTAX 78 by George Clisham (16K Level II) - Save time & money on taxes - 6 programs - completes form 1040 & schedules A,B,C,D,SE.
Cassette $9.95
Diskette version on 2 diskettes $17.95

MEMORY TEST PROGRAM by Software 80 - Checks out RAM, ROM, video RAM area & graphics functions.
Diskette $15.95

RSM MONITORS by Small System Software - numerous functions include - memory test - read & write machine language tapes - enter & execute machine language programs.
RSM-2 (16K Level I or II)... $26.95
RSM-2D disk version - includes read & write sec.... 29.95

DCV-1 by Small System Software - a disk conversion program for Level II machine language tapes such that the programs can be saved & loaded from disk. **$9.95**

MICROCHESS by Peter Jennings - machine language (4K Level I or II) - play chess against your computer - offers 3 levels of difficulty - includes instructions... **$19.95**

BRDIGE CHALLENGER by George Dulsman - 16K, Level II you and the dummy play against the computer in regular contract bridge - complete instructions... **$14.95**

AIRAID by Small System Software - a real time shooting gallery for the TRS-80 - player shoots cannon as airplanes fly by & parachutes land at adjustable speeds - requires the skill of an arcade game... **$14.95**

STAR TREK IIf by Lance Micklus - (16K Level II) advanced Star Trek - object is to explore as much of the galaxy as possible, destroy 20 Klingons and locate planets... **$14.95**

PILOT by Bob Edison - NOW even a young child can program the TRS-80 - a machine language version of the educational language PILOT with all the features of TINY PILOT & more... including a built in editor, sample programs included. **$14.95**

INTEGRATED FULL SCALE BUSINESS PACKAGE by Software 80
General Ledger - trial balance, income statements. $99.95
Invoicing - free adaptation to your invoice format. 99.95
Inventory Control - 2500 items per drive. 125.00
Accounts Receivable - w/aged receivalbe listings. 99.95
Accounts Payable - w/aged payable listings. 99.95
Payroll - prepares data for all tax forms. 99.95
Utilizes 2-4 disk drives. Entire package 495.00

The CPU Shop

The CPU Shop, Dept. N
39 Pleasant St.
Charlestown, Ma. 02129
Call 617/242-3350

TERMS: CHECK W/ORDER, M/C, VISA ADD $2.00 FOR SHIPPING. MASS. RESIDENTS ADD 5%

IS THERE A MODEM IN YOUR FUTURE?

T.R. Dettmann, Associate Editor

In this article, Terry discusses modems in general and his experience with the Control Data Computors at the University of Washington in Seattle.

The heydey of the mini- and micro- computer is here. No matter where you look, everyone is now talking only about the tremendous versatility of the small systems. Large computing systems are even spoken of by some in terms of their replacement by small systems. Despite the talk, it's a cinch that it won't happen soon. When, for example, was the last time you had access to 400K direct memory storage as well as so many Megabytes of disk and tape storage that you thought it was infinite? When did you last have available nearly any language available for your programming as well as sophisticated real time editing, text formatting routines with all of the nice to have features only available on large systems, or a version of Star Trek that had all of the features of the bridge of the Enterprise and more? All of this is available through time sharing on the large systems, and with the introduction of the Radio Shack telephone interface, you can make use of the capabilities of large system time sharing.

To use the telephone interface, you need to have at least a 16K Level II machine with expansion Interface. You then have to add an RS232C interface and a driver program. With this equipment (and an account at whatever computer you want to time share on), you too can make use of the capabilities of large computer systems. Let's look at the equipment you need to add to your system to see how each piece works.

THE RS232C INTERFACE

You can think of the RS232C as a magical black box that speaks "TRS80" to your computer and whatever is appropriate to the large computer you are connecting to. But that Isn't really what happens there at all. Actually, all the RS232C does is provide a programmable circuit that will reformat Incoming and outgoing data from your eight bit parallel bus inside the computer to make it correct for transmission on a two wire phone line. Other circuitry in the interface controls the number of bits transferred per second (the 'Baud' rate) the number of bits per character, and information necessary for matching to a particular computer system. All of this can be under direct control of a program executing in the computer.

Other types of interfaces are possible, this discription applies to the Radio Shack model specifically, but many function in this way.

THE TELEPHONE INTERACE

Interfaces such as the Radio Shack model I or II are often called either 'MODEMS' (for modulator-demodulator) or Acoustic Couplers. Each name is descriptive in its own way of the function of the unit. The interface accepts a standard phone receiver in a set of rubber baffled openings, one of which is a microphone and the other of which is a speaker. The interface accepts data transmited from the RS232C and uses it to modulate a carrier that will transmit the data over the phone line to a demodulator on the other end. It also demodulates incoming data transmissions and sends only the bit data to the RS232C.

There are two kinds of MODEMS. The type I MODEM is called an 'originate only' MODEM. It can only call in to another computer to carry on a computer to computer conversation, it can never receive a call from another computer (this does not mean that it can transfer data only one way, it can handle a two way coversation with another computer, It just cannot "answer" the phone). The type II MODEM is an 'orlginate and answer' type. This means that It can both make and receive calls. It Is also a little more expensive (about $50).

Most users don't need the type II MODEM. Since most use Is calling in to another computer, the type I is sufficient. A small business would find a type I MODEM would probably meet all of its needs If it kept some files on a large computer and its immediate files on a TRS80. A type II MODEM would be necessary on at least one of two TRS80's that are to be tied together (for example, you could play a two person real time Star Trek by connecting two TRS80's together with a set of Interfaces and running the respective portions of the program in each computer).

THE DRIVER PROGRAM

Having an Interface Is not enough, something has to translate the ASCII codes coming In and send the data out at the proper rate. Luckily, a program to do that Is included on tape with the RS232C

interface. The program (in Assembly Language) is also given in the handbook. Unhappily, this program makes your TRS80 what is known in time sharing as a 'Dumb Terminal'. Your TRS80 is now only a keyboard and a screen interfacing with the phone line. You can reprogram it yourself (instructions are included) but you had better be good at Assembly Language programming. No other terminal programs are on the market at this time though they are sure to come out in the near future, and will allow you to send data from your TRS80 to a large computer or vice-versa.

HOW TO USE A MODEM

The drawing on the front cover shows an actual session with a CDC system. Once you have opened an account with the system you want to use (yes, they're going to charge real money), you can call the computer on the phone. Once you have the other computer, you use whatever access procedures are required to get on; such as account numbers, passwords, or special codes, and then you're on. It doesn't feel different at first, but the prompts will be new and when you try to do something you are used to, the computer will probably ask what you mean. Once you are on, you are using your own money, so you should be familiar with how the other system works and have a plan for what you want to do. You can now program in any language the other computer supports (PASCAL anyone?) or access special routines not available on your system. Even more impressive is the fact that though you are talking to the other computer with only eight bits per charcter, it will be doing your calculations with whatever bit size is appropriate for that computer. The CDC for example has 60 bit words compared to the eight for the TRS80. This gives a definite improvement in the accuracy of any calculations you may have to do.

AN EVALUATION

I have found through use that the Phone interface and RS232C are useful and well built. The RS232C fits nicely into a space made for it in the top of the expansion interface. In fact, installation is so easy that you do it yourself without voiding your warranty. Some of the older expansion interfaces don't have the required connections in the space provided and so a Radio Shack Service Center has to install the connector. Be careful though in putting it in since it turns out that there is no way to align the board properly except by trial and error. Once in, any movement of the expansion interface will cause the RS232C to shift out of position. When this happens you get funny screen displays. You only need a small phillips screw driver to install the board and a washer for one side where the hole is too big. Once the board is in, the next big problem is setting the sense switch block which determines parity, baud rate etc. The manual gives a setting which is good for most phone interface work, but check with your computer center to be sure you have the right combination. With eight switches, random selection will take a long time before you find the right combination.

In actual day to day use for about a month since my unit arrived, the only probelm I have encountered has been an annoying tendency to drop out on long terminal sessions. After working for a half an hour to 45 minutes, I would occasionally get back error messages that indicated that the system had received a bad character though I could see that I had typed the correct one from my screen display. Within 5-15 minutes after this starts, the terminal usually drops off the line altogether. On some systems this can be catastrophic to say the least, however, this type of behavior is common among Acoustic Couplers. If the system you tie into doesn't protect your files if you drop off accidentally, then do everything with care and backup your files every chance you get.

PRICES

In any large city, there are bound to be a number of time sharing services listed in the phone directory under 'Data Processing'. If you call around to each you will be able to get the best deal. Every service is different, some are for specialized markets such as CPA's or Engineers. Even pricing schemes vary. If you are lucky, and a state University Computing Center is available, you may be able to get a good rate there. Some commercial services require a minimum monthly charge (generally in the range of $100 to $500). Services with such minimums are generally too expensive for the small user. Services that don't ask a minimum are more suitable for the small user, but be sure the pricing for time used is clear. It is possible that you could spend more for such a service than you would for one with a minimum. Most services price by resource units or some other unit which is tied to CPU time used. Memory used is adjusted to the common units as is time for tapes and disks. Be sure you know what you are getting into when you agree to a particular service. It may also be possible to get special deals with local clubs. One computer club in Seattle has an arrangement for after hours time on a commercial service's computer. Since they get billed as a club, their rates are exceptionally low.

Computer time sharing is not for every TRS80 owner, nor is it necessarily cheap. But if you have a need for a larger computer and you like working at home, time sharing might be a realistic option for you to consider.

...80

NOW THE TRS-80 HAS FORTRAN

T.R. Dettmann, Associate Editor

MICROSOFT, the people who gave us Level II BASIC, have come through again, this time with a disk based FORTRAN system for the TRS80. If you need to use the power of FORTRAN, or you want to use a language that has a compiler instead of an interpreter, or if you want a MACRO assembler, then MICROSOFT's FORTRAN system could be for you.

The system runs on a 32K machine with 1 disk as a minimum configuration and it sells for $350. This is pretty steep for most users, but if you can afford the upgrade on your system or if you already have enough of a system to make it work, then it is an excellent addition to your programming library.

Let's look at what FORTRAN is and what it can do for you.

FORTRAN is a high level, machine independent programming language that was developed by IBM starting in 1954. It made its first appearance in 1957 on the IBM704 computer, and rapidly became the standard language for Engineering and Science calculation. It is one of the most popular languages to ever run on any computer. It can be found on nearly every large computer system because of its inherent flexibility.

BASIC was developed in 1965 by two college professors as a simplified form of FORTRAN for those who didn't have the time or the need to learn all of the complexities of the FORTRAN language. Thus in a way, the TRS80 Level II BASIC is a grandchild of FORTRAN and very closely related. Other languages have been developed to replace FORTRAN, but none has been able to convert the users of FORTRAN. Languages such as PL/1, ALGOL, and APL were all supposed to replace FORTRAN, but none has. More recently, there has been a move to push PASCAL as the replacement for FORTRAN.

Computer Science graudates all love it and most of the micro-computer magazines have all at sometime praised it as the best thing since sex, but FORTRAN is so deeply ingrained in Engineering and Science that it promises to be around for years to come. What's more, FORTRAN programs have been developed to do nearly anything that one could want to do with a computer. These routines are often available in various books and other publications. My own library of programs has several hundred routines accumulated over many years of programming large computers. Now I can use them directly on my TRS80 without translation on BASIC.

BUT WHAT IS A COMPILER?

TRS80 BASIC is provided in ROM as an interpreter. This means that the BASIC program you write is read one statement at a time by the interpreter and run as it is read. This is a slow way to run a computer since a statement must be read and translated each time it is executed. This means that a statement in a loop with 1000 iterations is translated from BASIC to machine language each of the 1000 times that the program goes through the loop. This isn't bad for program development, but it wastes computer time when you are running the program for production.

It would be nice if after a program was checked out, it could be translated only once to machine language and then run that way. Execution would be faster since the computer would not have to redo each statement everytime it was used. Well, this is what a compiler does.

Programs written in FORTRAN are read by the compiler and converted to machine language immediately. Then when the program is executed, it runs just like any other binary program.

On large machines, compilers are the normal way that languages are used. Only on small machines are interpreters common. Once the machine language code is generated, it can be saved and run over and over again as needed. This can be a great convenience as well as a real timesaver.

MACROS

The MICROSOFT FORTRAN also includes a complete Editor-Assembler package as a part of the system. If you write in Assembly language you will find this interesting. Most significantly, the Assembler also includes a full MACRO capability. For those who are not familiar with machine language programming, a MACRO is a sort of subroutine in Assembly language with a special twist. Instead of being called like a BASIC subroutine, a process which takes memory space to save register values and addresses, a MACRO inserts the procedure defined directly into the program at the point called without any special external references. In this way, routines that are used frequently can be defined once, and then inserted into the program with a single reference. This is a far more efficient procedure. than using a subroutine in the same circumstances.

Now let's look at MICROSOFT's FORTRAN to see what features it has.

MICROSOFT FORTRAN

The FORTRAN package comes complete on disk with the FORTRAN compiler, a MACRO assembler, and a loader that can load programs generated by the system at any place in memory you desire. This last feature is particularly nice since an object code module produced by the compiler can be put at any location and automatically linked to a subroutine library which you can create with subroutines you have already checked out on previous runs. This type of code is called 'relocatable' since the actual memory locations are only decided upon by the loader itself.

The FORTRAN provided is an industry standard FORTRAN (ANSI X3.9-1966) except that the ability to handle complex numbers directly is not available. A number of enhancements are also provided which extend the utility of the system. Some that Level II BASIC users will appreciate are the ability to do mixed mode arithmetic (integer and real mixed in the same expression), End of File and Error transfer within the program, and the ability to use Hexadecimal constants in the program. Custom drivers can also be interfaced to FORTRAN so that you can use non-standard equipment with your system.

Of even greater interest, is the ability to create a library of subroutines that are always available to the loader. In this way, subroutines can be checked out to do calculations you need often, and then you may simply call them in your program, the loader will see that the routine is automatically attached. The system provides 49 routines already to do things like absolute value, double precision functions, and much more. This one feature alone is worth its weight in gold if you do many calculations that use the same subroutines.

MICROSOFT's new FORTRAN package for the TRS80 is one of a line of such packages that MICROSOFT makes for 8080 and Z80 machines. As such, it has been well checked out and is sure to perform well for the user. Before going out to buy the system though, you should realize FORTRAN is not as forgiving as BASIC. You have to be more careful with your specifications when writing a FORTRAN program, but FORTRAN is also more powerful. If you like FORTRAN, or you would like to take advantage of the routines already in FORTRAN, then this package will be the answer you are looking for.

In the next issue, we will evaluate the FORTRAN package, including the compiler, the assembler, and the loader.

...80

MICRO-COMPUTERS IN THE FUTURES MARKET

Jeffrey A. Miller, Eugene, Ore.

The most interesting aspect of computers in general, and of Microcomputers in particular, is their application to many other fields of endeavor. In this article, Mr. Miller not only tells about Futures Trading, but his plan of attack in switching over from another system is worth note. This is especially important when considering uses for your computer where "your money" is on the line. In this case the guide lines described by the author are worth keeping in mind.

With the value of the United States economy exceeding $2 trillion dollars isn't it surprising how little is known about the "Futures Market"?

"Why?", you may ask!

Well, how about the little known fact that the value of all the futures market trading done in 1977 was an astounding $1 trillion dollars! Yes, only half the value of the entire US economy. Not bad for the perennial black sheep of the investment world. In fact, futures markets often trade a greater dollar value of goods than the value of the stocks traded on all stock exchanges.

WHAT IS THE FUTURES MARKET?

The "futures markets" I refer to are also called "commodity exchanges" or "commodity futures markets" and exist all over the world. There are eleven separate commodity exchanges or futures markets in the United States alone. Plus there are several in London, Hong Kong and Sydney, Australia. Although the term "commodity" refers to any good that is bought and sold, we are concerned only with those "commodities" traded on the organized exchanges. These range from the grain and meat (wheat, corn, soybeans, cattle, hogs, etc.) markets in Chicago to the food (coffee, cocoa, sugar, potatoes, orange juice, etc.) markets in New York to the international commodities traded all over the world, and back to Chicago and the financial (foreign currencies, gold, silver Treasury Bills, etc.) markets. There are also numerous smaller exchanges located in the US and other parts of the world where significant trading takes place in the grains, cotton, wool, sugar, coffee, cocoa, gold, silver and so forth.

The commodity futures markets developed for several basic reasons. One, like any market anywhere, the commodity futures markets set prices for their particular commodities. Farmers, elevator operators, bakers, packing houses, and other individuals can then use those "market prices" to set their own individual prices after adjusting the "price" to reflect quality differences, location differences, and time (of delivery) differences. A second reason for their development is that, like stock exchanges, a commodity futures market is a mechanism to provide capital to the farmers, ranchers, bakers, packers and other participants. The market is only a mechanism because the capital (money) actually is obtained from the many speculators who must be present if the market is to operate efficiently.

Commodity futures exchanges operate much like the stock exchanges we are all familiar with. In the stock market tho, people buy and sell,...well, stocks! Of course there are warrants, options and bonds and government securities, etc. In the commodity futures markets, though, people don't buy and sell "commodities"! Rather, they buy and sell contracts, which require at some time in the future one person (the seller) to deliver a commodity and the other person (the buyer) to accept delivery and pay for the commodity. The two key words are. "future" and "contracts".

Buyers and sellers in the commodity futures markets actually buy and sell "futures

contracts". These contracts are real, legal contracts which can be enforced by the courts.

However, since each futures contract for any particular commodity is exactly like every other futures contract for that particular commodity except for the price, the contracts are easily exchanged. That is, if you buy a futures contract today, you can easily sell that contract tomorrow, next week, or even later today. Any change in the price which ocurred between the time you bought a futures contract and the time you sell that contract represents your gain or loss, depending on the direction prices moved. It should be understood that this explanation of a commodity market is greatly oversimplified but the key concepts are here. Participants in the market generally refer to this activity as "futures trading", or as trading in the "futures market". This way it is clear they refer to futures and not the "actuals" or "cash" commodity markets where the physical commodities are bought and sold. There is a definite relationship between the cash markets and the futures markets but it is a dynamic relationship and not too easily defined.

Finally, since futures trading is actually trading "Futures contracts" it is just as easy to originally "sell" a contract and "buy" a contract and "sell" it later. The difference being, if you sell first, prices must decline in order for you to make a profit. If you buy first, prices must rise in order for you to make a profit. The idea then is that as a speculator you attempt to determine the probable direction of the prices and then buy or sell futures contracts, depending on the direction. Therein lies the rub. How does one determine the probably future direction of the prices? Unless you can forsee the future, the only alternative is to examine information about the markets to try and determine the most probable direction of price movement. This then, is the crux of the matter. What means are available to allow individuals to examine large amounts of information quickly, easily, and accurately? The best answer to date is the "Micro" or personal computer. It might be useful to quickly review the development of the personal computer and examine its application to futures trading.

ENTER THE MICRO-COMPUTER

Some years back the electronics industry developed what they call the "microprocessor chip". These chips are the brains of all microcomputers. Not only have they reduced both the size and cost of micros but they make it possible for anyone to run one! In fact, you can purchase one for less money and learn to use it in less time than is required for many recreational activities. The system this article is based on cost less than $1000 and yet does everything described there and much more!

What are the different kinds of micros currently available? The earliest were kits you had to put together. They generally appeal to the computer hobbyist. Today, there are several brands of "ready to run" computers. These include the TRS-80 (Radio Shack), the Commodrore PET and Processor Technology SOL. My own, the one on which this article is based, is a TRS-80 with Level II and 16 K Ram.

FITTING THE MICRO TO THE TRADER

Generally the first step in choosing a system is determined by the traders needs. That is, how many commodities to follow and what data will be recorded for each?

The main point to be considered here is that one of the most expensive areas of any micro is in the memory. For this reason the trader should purchase a system which has only enough memory to do the job. There is an obvious trade-off here between time and money. The more money (memory) the less time it takes to perform a series of operations on large amounts of data. Our objective is to strike a balance, sacrificing some time in order to save considerable money. The most important point here is to purchase a system capable of expanding as the individual's needs, sophistication & money increase. All the systems mentioned are expandable.

THE SOFTWARE

Once we've assembled our hardware (the computer and its peripherals-video screen, keyboard, cassette recorder, etc.) we turn our attention to the software. Software is computerese for "programs" or lists of step by step instructions for the computer. Without software even the most expensive computer is nothing but a hunk of junk. There are only two methods available to obtain software--write your own, or buy some! If you have the time and ability you can develop your own. Most of us though are better served by purchasing programs developed by skilled programmers. One such software package is the Commodity Market Analyst 1. or CMA1. published by MicroFutures Trading Co.

Software is also available from Commodity Concepts and others; if you buy the Comm Basic system it comes complete with software. CMA1. allows each trader to develop and test their own unique strategy. I simply do not believe anyone will sell a highly profitable "system" much less put it on tape for a micro. If they did, how profitable would it be when thousands of traders use it? A check of the ads in various computer or commodity magazines will indicate several individuals and firms, including those above, which sell packaged software or will develop custom software.

THE METHOD

Regardless of how you obtain your software there are still some important steps prior to our research efforts. These steps are 1) "debugging" the

(Continued on page 17)

software; 2) "double checking" calculations and graphs; and 3) running the micro along with your old P & P (pencil and paper) method. The first step, "debugging", applies primarily to individually developed software or custom software. It is a technical term and usually refers to checking each possible routine to insure proper operation. Generally, debugging should not be required on packaged software.

The second step, "double checking" is valuable for two reasons: one, it insures that calculations are correct and results are what the user expected; two, it allows the user to become thoroughly familiar with the operation of the programs. Double checking is done by simply "running" the program and entering data as required. Then the user compares the computer's results with those obtained using P & P and a hand calculator. The key to both debugging and double-checking is to use "phoney" data. The final step is the most important and requires the most effort. It involves both your new computerized system and your old P & P method. Ideally, you should use both methods for several months under actual operating conditions. Using actual data the trader will be comparing computer results with P & P results. Once the trader is satisfied that the computer system is functioning properly and that the trader understands the computer's results then they can cease the P & P method. Be realistic, please! Don't perform step three following any more than one or two different commodities. Once the trader concludes step three expansion to any number of commodities is possible.

DATA BASE CONSTRUCTION

Now, we can construct our data base. A data base is merely an organized collection of information. For commodity traders the data base might include any number of commodities and contracts, prices, volume, open interest and a variety of fundamental information. YES, FUNDAMENTAL!! Just because we use a computer doesn't make us technical traders. Fundamentals can be stored, tracked and analyzed using micro's. "Fundamentals" refers to supply and demand conditions while "technicals" referes to prices, volumes, open interests and their various actions.

In developing a data base we bump into the same considerations we encountered in determining memory size, money and time. It is possible to build your own data base if your software is so designed. This method saves money but costs time. On the other hand some firms may sell data tapes as well as packaged software. Buying your data base obviously costs money but it does save time. A third alternative is to buy a software package like CMA1. which allows you to build your data base plus update it as often as necessary. Then, by purchasing a selected number of historical data tapes and continuously updating them you have the most cost-effective method of building a very valuable data base.

It takes considerable effort to obtain the proper hardware and software, check them out and then buy or build a data base. But now we can direct our efforts into "research" designed to develope or improve a profitable trading strategy. That's a scary word to most of us, "research"! How do we do research and what does it mean? I will refer the reader to any general book on futures trading and will recommend THE COMMODITY FUTURES GAME: WHO WINS? WHO LOSES? WHY? by Tewels, Harlow and Stone and the more recent COMMONDITY TRADING SYSTEMS and METHODS by Kaufman. Check your local library or bookstore.

For this article I will only list some of the research features available using the software package, CMA1. and a TRS-80 Level II with 16K Ram. Generally the user can examine the data base using price charts, price lists, spread charts and lists, ratio charts and lists, trend lines on any price, spread or ratio chart, moving averages (any number of days) on prices, spreads or ratios in chart or list form, point and figure charts with box size and reversals set by user, percentage filters on trends or moving averages and more. The programs take the user from start to finish in performing these operations. The program does not tell you what the information means or how to create a profitable strategy. The trader is presumed to know what a moving average is or what a point and figure chart is. The ultimate value of CMA1. and other packages like it is that they allow the trader to study more than one commodity in depth. In fact they allow the trader to study any number of commodities very quickly, easily, and accurately. (Hint - diversification!) The computer does not transpose numbers, mess up a calculation, mis-read a chart, fall asleep, or get indigestion.

The astute trader will use his/her computer and its software for number crunching and will record those results using pencil & paper. A group of results can then be used to determine a profitable strategy or eliminate an unprofitable one.

In the old, wild West the Colt '45 was known as the "great equalizer". The microcomputer of today is the new "equalizer", at least in futures trading. At last the small speculator has the same kind of tools available to them as the big speculator has long had. All that is required is to use your micro's wisely, follow sound money management, and then combine the two into a profitable trading strategy. ...80

THE BETTER BYTE BOX

Ray Thompson, Pearl City, HI

If you have Level II, you are familiar with trying to load tapes that give little stars in the upper right of the screen - and the left one turns into a "C", or they both remain on. So you rewind your tape, change the volume setting slightly and try again, and again!

If you save one of your programs and the tape is good, you can usually reload it without problems. But when you try a tape made on another system, or you try to load system tapes (like the program conversion tape) you have trouble!

Being quite dissatisfied with this situation, I finally hooked up the oscilloscope and found large overshoots and huge amounts of ripple and hum. Disregarding what the manufacturer designed into the '80 to obtain data from pulses like these, I tried a small RC filter with a diode to clean up the waveform. It loaded the offending tape on the very first try! Fantastic, right? Well, no, not really...I found that tapes generated on my own system would not load now. So out came the oscilloscope again, and I found that the polarity needed to be reversed for my tapes. So I installed a reversing switch around the diodes and tried again. By paying attention to the polarity, all my "hard to load" tapes loaded fine.

Now hooking an oscilloscope to the filter just to load *a* tape just doesn't make a lot of sense. So I added a meter to give the necessary indications. Now I have a little box with the filter, the switch and a meter. Just plug the box into the earphone output of the recorder and the TRS-80 into the box. Turn the tape on manual, flip the polarity switch and leave it in the position which gives the highest reading. Adjust the volume level of the recorder for 15 microamps, rewind the tape and watch that hummer load!

I built up my box and went around to several other owners in the area and it worked great. One friend had the in-Memory programs which were always hard to load. Now they loaded on the first pass. It should be noted that this will do nothing for tapes with "drop outs".

Sweet success? For a while anyway. I finally ran into one that would not load. Putting the oscilloscope on it again indicated that it needed a much larger signal than was there. It was at this point that I changed the RC filter and eliminated the capacitor, and it appeared to clamp the overshoot better. Then I happened to put pressure on the right front corner of the cassette, and WOW!!, the amplitude of the signal jumped up where it should be! This seemed to be an indication of head alignment differences between the originating tape recorder and the one playing it back. Carefully looking at this new addition to tape loading procedures, I found that the tape cassette can actually twist a little from beginning to end due to differences in torque. This could explain why programs at the beginning or end might be easier to load. If the head is aligned with the tape, the amount of torque does not make any noticeable difference. When they are not together, there is a whole lot of difference.

Now the second modification: Take the bottom of the cassette recorder off first, then remove the top cover and drill a little hole in it so that the head alignment can be done thru the hole with a small screwdriver. Then decide which tape you will use as a standard. Most likely it will be one of those made on an expensive machine. I decided on the Program Conversion tape. Now if you get your old "C's" back, just slip your small screwdriver down the slot, watch the meter and peak the head into maximum reading. If the tape does not have actual "holes" in it, it will load for certain.

One other modification you may wish to do to remove ground loops, is to cut the circuit board connection between the AUX line return and the MIC return. Then add a jumper wire from the AUX line return to the EARphone return.

If you are going to put any kind of a "box" between your TRS-80 and the cassette recorder, put in one that will actually do something!

..80

(Editors note: Since this seems to be a problem for most of us, we will include Ray's complete address. If you care to correspond with him, please include a stamped self addressed envelope, as we well know, answering a few hundred letters can be costly and time consuming. Ray Thompson, 2040 Komo Mai Drive, Pearl City, Hawaii 96782)

PARTS LIST

2 Germanium Signal Diodes
2 10K 5% Resistors
1 1K Resistor
1 82 Ohm Resistor
1 Double pole double throw Switch
1 0-50 Microamp Meter (RS Micronta Cat #22-051 or equiv
1 Miniature Jack
1 Miniature Plug
1 2" X 2" circuit board to mount parts
18" shielded wire
18" tie wire

Box for parts & meter - Archer Cat #27D-233 or equiv

Note: Use of Silicon Diodes will work, but give signal saturation at 1D microamps instead of 15 microamps.

OPERATING INSTRUCTIONS

1 - Flip switch to determine highest reading.
2 - Leave switch at this setting adjust volume for 15 microamps.
3 - Rewind tape to beginning and Load it!

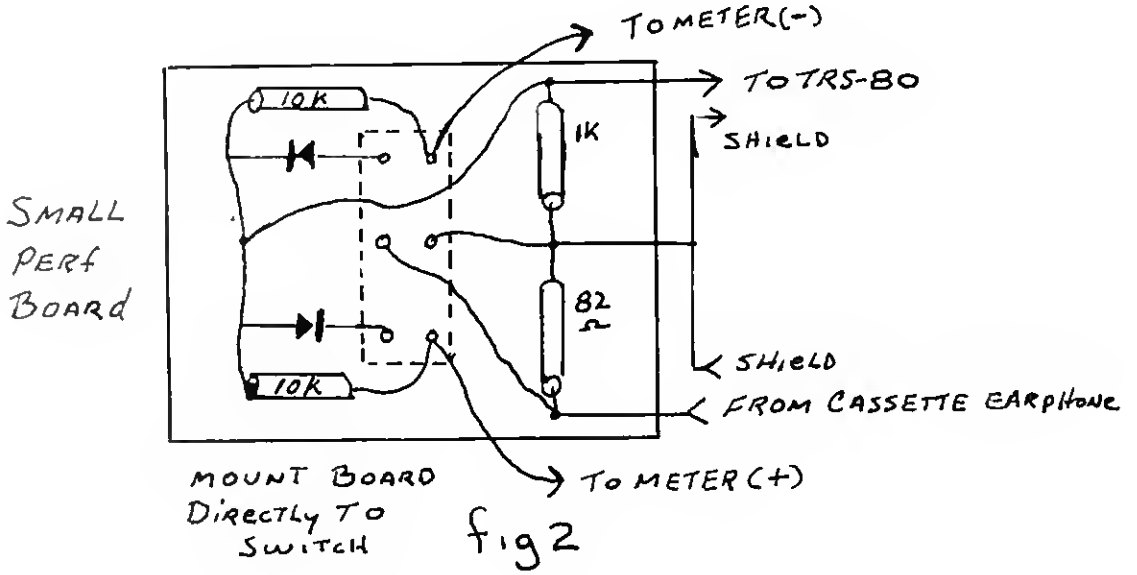


fig 2

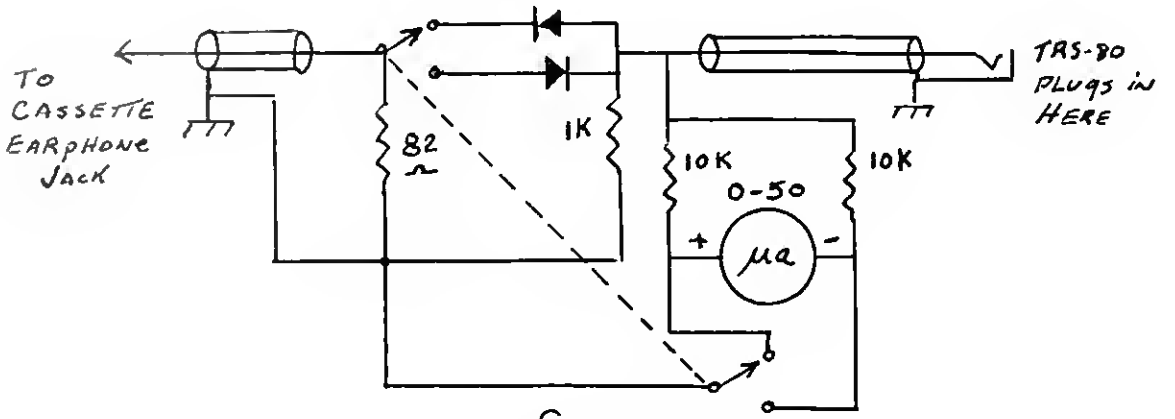


fig 1

SYSTEM/COMMAND

Phil Pilgrim

When someone talks about black and white photos or black and white TV, you take for granted that various shades of gray are also involved. So, why should black and white graphics on the TRS-80 be just black and white? There is no reason I could think of; so here is a program for getting those dirty charcoals, dingy grays, and white-but-not-brights. The principle is simple: Put a camera in front of the screen, and open the shutter. Those areas to be brightest are displayed longest; dimmer areas, shown for less time. Close the shutter, develope the film, et voilal Gray level graphics!

The assembly program shown here (GRAY) can be called from Level II basic. It requires the BASIC program to POKE a set of intensity values (0-127) into a 1024 byte memory array, each array position corresponding to one character position on the screen. When GRAY is called, it cycles through the intensity vector and causes the corresponding character cells on the screen to be lighted for a time period proportionate to their intensity values. It then will blank the entire screen. The result, if a film were exposed to the screen, would be a set of rectangles having various shades of gray.

```
 90 REM TEST PATTERN PROGRAM; SET MEM SIZE TO 31660.
100 FOR I=31744 TO 32767:POKE I,0:NEXT:REM BLANK VBUF
110 FOR I=0 TO 63:FOR J=I TO I+511 STEP 64
120 POKE 32000+J,I*2:NEXT:NEXT:REM POKE INTENS. IN VBUF
130 FOR L=0 TO 4:T=2↑L:CLS:PRINT @960,"FRAMES:";T
140 K$=INKEY$:IF K$="" THEN 140:REM WAIT FOR KEYSTRIKE
150 FOR I=1 TO T:K=USR(0):NEXT:REM FLASH PATTERN T TIMES
160 K$=INKEY$:IF K$="" THEN 160:REM WAIT FOR KEYSTRIKE
170 NEXT:GOTO 130:REM KEEP LOOPING
```

BASIC test program for GRAY (16K, Level II).

```
DELAY   EQU   0060H  ;Delay routine in ROM
VIDEO   EQU   3C00H  ;Display memory
        ORG   7DB5H  ;For 16K:MEM SZ=31660
START   LD    HL,GRAY;Set up USR addr.
        LD    (16526),HL
        JP    1A19H  ;Back to BASIC
GRAY    XOR   A      ;Set max to 0.
        LD    HL,VBUF;Intensity vector
        LD    BC,1024;Size
MAX     CPI          ;Compare (HL) w/ A
        JP    P,AGAIN;Smaller
        DEC   HL
        LD    A,(HL) ;New max.
        INC   HL
AGAIN   JP    PE,MAX ;Back for another.
        OR    A      ;Max=0?
        RET   Z      ;Yes: Return.
DLOOP   LD    DE,VIDEO-VBUF-1
        LD    HL,VBUF
        LD    BC,1024
FIND    CPIR         ;Find 1st (HL)=A
        JR    NZ,DWAIT;No more
        PUSH  HL
        ADD   HL,DE  ;Corresp. VIDEO addr.
        LD    (HL),191;Light it up.
        POP   HL
        JP    PE,FIND;Back for more.
DWAIT   PUSH  AF
        LD    BC,256
        CALL  DELAY  ;Wait awhile.
        POP   AF
        DEC   A      ;Next intens. val.
        JR    NZ,DLOOP;Do it again.
        LD    HL,VIDEO
        LD    (HL),128;Blank 1st position.
        LD    DE,VIDEO+1
        LD    BC,1023
        LDIR         ;Blank the rest,
        RET          ;and return.
VBUF    DEFS  1024   ;The intensity vector.
        END   START  ;For autostart.
```

The assembled version of GRAY shown here has four sections. The one labeled START puts the address of GRAY into the USR transfer vector and jumps back to BASIC.

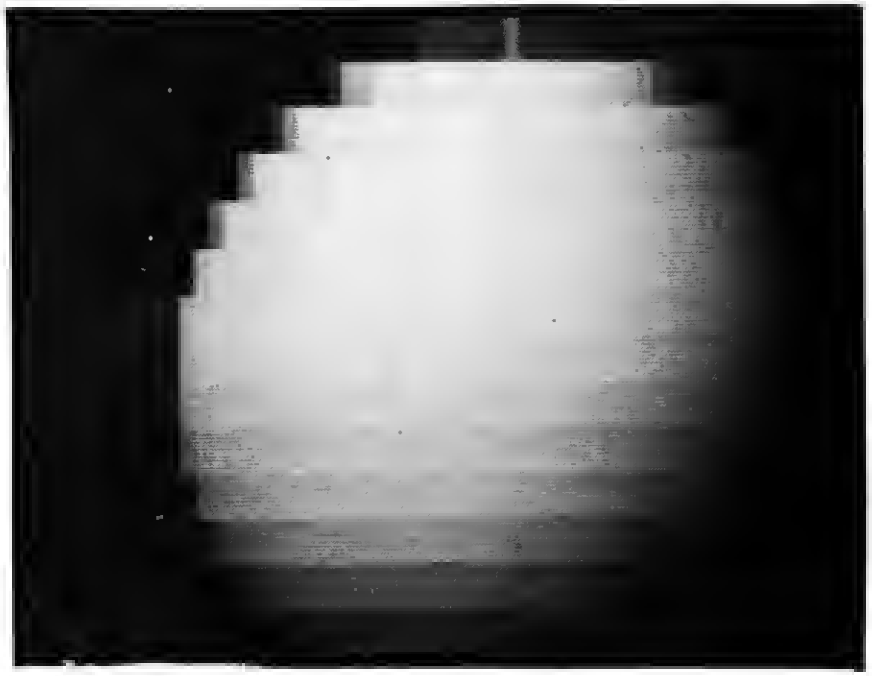
Since START appears in the END pseudo-instruction, the program will autostart here after loading through SYSTEM. Just type / and ENTER. The section starting with GRAY (the entry point) and ending just before DLOOP scans the intensity array (VBUF) for the largest intensity, returning it in the accumulator.

The section starting with DLOOP and ending with the JR NZ, DLOOP decrements the intensity value in A until it reaches zero, cycling through VBUF each time and lighting those positions having intensity equal to A. Those stay lighted until after DLOOP is finished. The final section blanks the entire screen (using LDIR) and returns.

To test GRAY, set your monitor brightness and contrast so no background lines are visible. Mark the positions of these

controls for future reference. Set up your camera on a tripod or camera clamp and focus on the screen of your monitor. (If you can't get close enough to fill the frame use a fine-grain film like Kodak's Panatomic-X and blow it up later.) Load the BASIC test program shown and turn out the lights. It has to be dark to eliminate reflections from the screen. RUN the program. When the screen shows FRAMES: n, open the shutter (using a cable release), and hit any key on the keyboard. The test pattern will cycle N times and stop. Close the shutter, wind the film, hit any key on the keyboard, and you're ready for the next exposure. Whenever FRAMES cycles back to 1, set a new F-stop on your camera and continue. When the pictures are developed, you will be able to judge which F-stop/FRAMES combination is best.

With GRAY and a little ingenuity, you should be able to create some remarkably realistic pictures. The photo of a sphere lighted from the above left shows what can be done in displaying solid objects.



...Phil

Tom Rosenbaum,
Technical Editor

VIEW FROM THE TOP OF THE STACK

Last issue you were left with the problem of building a program which would print out your name on the screen. Listed below is a program which will print the name "JOHN DOE" on the top line of your screen.

The program is not elegant, but it uses only those instructions which were explained last issue:

```
3E 4A     LD   A,4A      ;ASCII J
32 20 3C  LD   3C20,A    ;store to screen
3E 4F     LD   A,4F      ;ASCII O
32 21 3C  LD   3C21,A
3E 48     LD   A,48      ;ASCII H
32 22 3C  LD   3C22
3E 4E     LD   A,4E      ;ASCII N
32 23 3C  LD   3C23,A
3E 20     LD   A,20      ;ASCII blank
32 24 3C  LD   3C24
3E 44     LD   A,44      ;ASCII D
32 25 3C  LD   3C25,A
3E 4F     LD   A,4F      ;ASCII O
32 26 3C  LD   3C26
3E 45     LD   A,45      ;ASCII E
32 32 3C  LD   3C27
```

Another way to accomplish the same job would be:

```
26 3C     LD H,3C ;load starting address
2E nn     LD L,nn ;into the HL register pair
3E 4A     LD A,4A ;ASCII J
77        LD(HL),A;store to screen
2C        INC L ;Increment the L register
3E 4F     LD A,4F ;ASCII D
77        LD(HL),A;
2C        INC L ;
3E 48     LD A,48 ;ASCII H
77        LD(HL),A
2C        INC L
3E 4E     LD A,4E ;ASCII N
77        LD(HL),A
2C        INC L
3E 20     LD A,20 ;ASCII blank
77        LD(HL),A
2C        INC L
3E 44     LD A,44 ;ASCII D
77        LD(HL),A
2C        INC L
3E 4F     LD A,4F  ;ASCII D
77        LD(HL),A
2C        INC L
3E 45     LD A,45 ;ASCII E
77        LD(HL),A
```

nn in the second instruction defines where the name will appear on the screen. If nn = 10, the name will start at the top left hand corner; if nn = 20H the name will begin at the middle of the top line. Remember that TBUG uses the left 10H characters of the screen.

This program is "relocatable", which means that it may be placed anywhere in free RAM and it will run. Free RAM means any portion of RAM not used by another program which may be in memory such as the Level I or II ROM, the DOS, T BUG, etc. It must also be in an area of RAM which is safe from encroachment by the stack. What the stack is will be explained in a later session. When T BUG is loaded into the machine, it will occupy RAM from 4380 to 4980 (Level II). All memory locations above 4980H (Level II), will now be usable for your machine language programs. You may modify the above programs to display different names by changing the ASCII letters which are printed on the screen.

16 BIT LOAD GROUP

There are 5 general purpose "working" or "scratchpad" registers: BC, DE, HL, IX AND IY. There is a series of instructions allowing for the transfer of data between these registers and memory.

These instructions are summarized below. In the explanation of the instructions, dd stands for any one of the register pairs BC, DE, HL or SP (stack pointer - a special register we will look at later), nn stands for any memory location from OOOO to FFFF and (nn) stands for the CONTENTS of the memory location whose address is nn.

1. LD dd,nn	nn is loaded into register pair dd
2. SL IX,nn	nn is loaded into IX
3. LD IY,nn	nn is loaded into IY
4. LD HL,(nn)	(nn) is loaded into HL
5. LD dd,(nn)	(nn) is loaded into dd
6. LD IX,(nn)	(nn) is loaded into IX
7. LD IY,(nn)	(nn) is loaded into IY
6. LD(nn),HL	HL is loaded into (nn)
9. LD(nn),dd	dd is loaded into (nn)
10. LD(nn),IX	IX is loaded into (nn)
11. LD(nn),IY	IY is loaded into (nn)

Care must be exercised when using these instructions because they use the now infamous reverse order loading. This means that when a register pair is being loaded into memory or vice versa, the high order byte (B of BC, D of DE or H of HL) goes into (or comes from) nn while the low order byte goes into nn + 1. For example, the instruction LD (4200),HL would be assembled as 22 00 42 instead of 22 42 00 which would seem to make more sense.

JUMP GROUP

One of the most common things that must be done in programs is the repetitive execution of certain instructions. Think of the number of times you use a FOR-NEXT loop in a Basic program. In an assembly language program you must build your FOR-NEXT loops from scratch. The NEXT part of the loop is accomplished by a JUMP statement. JUMP statements transfer control of the program from the next instruction to another location in memory by altering the contents of the Program Counter (PC). There are two types of jumps: relative and absolute.

The relative jump will allow you to jump a certain displacement (e) from the current address contained in the PC. The range of this displacement is + 129 to -126 (decimal). An absolute jump forces the PC to the address supplied by the instruction. The big benefit of the relative jump is that it only takes two bytes, while the absolute jump takes three bytes. Another advantage is that programs written using relative jumps will work if they are moved to any part of memory. Programs written with absolute jumps will have to have the jump addresses changed if they are to work when moved.

Another type of JUMP instruction will allow the jump to be conditional upon the state of the flags in the flag (F) register. The flag register contains information on five flags: 1. Zero Flag, 2. Carry Flag 3. Sign Flag 4. Parity Flag and 5. half Carry Flag. The execution of an instruction may effect all, some or none of these flags so be careful to check your Z80 reference manual to see how an instruction will effect the flag before using a conditional based on the state of a flag. the JUMP instructions are summarized below:

1. JP nn	jump to eddress nn
2. JP cc,nn	jump to nn if condition cc is true ·

1. JP nn	jump to eddress nn
2. JP cc,nn	jump to nn if condition cc is true - the ellowable true velues for cc ere: NZ - zero fleg = O, NC · cerry fleg = O PO · perity flag = O, P · sign fleg = O Z - zero fleg = 1, C · carry fleg = 1 PE · perity fleg = = 1, M · sign fleg = 1
3. JR e	jump relative by the displacement e (+ 129 to -126)
4. JR C,e	jump reletive by e if the cerry fleg = 1
5. JR NC,e	jump reletive by e if the cerry fleg = O
8. JR Z,e	jump reletive by e if the zero fleg = 1
7. JR NZ,e	jump reletive by e if the zero fleg = O
8. JP HL	jump to eddress in the HL register pair
9. JP IX	jump to eddrees in the IX register
10. JP IY	jump to eddress in the IY register

There is one additional, powerful, instruction, the Decrement and Jump if NOT Zero (DJNZ). This instruction will decrement the B register and jump by a displacement of e if there is a non-zero value left in the B register after it is decremented. If the B register is equal to zero the jump will not occur.

As a practical example of the DJNZ instruction let's try to make a shorter program for displaying names on the screen. This program will assume the name we want to store on the screen is stored in continuous memory locations. We will set the HL pair to point to where the name is stored and set the DE pair to point to the area of the screen where we want the name. Then we will set the B register to the number of characters in the name.

```
4000 06 08      LD B,8          ;# charecters in name
4C02 21 00 4D   LD HL,4Doo      ;neme stored here
4C05 11 10 3C   LD DE,3C10      ;displey eddress
4C08 7E         LD A,(HL)       ;get hcer to A register
4C09 12         LD(DE),A        ;store to screen
4CoA 13         INC DE          ;next cherecter
4C0B 23         INC HL          ;next screen tocation
4C0C 10 F2      DJNZ,-12        ;loop until done
4Doo 4A         DS 1            ;J
4D01 4F         DS 1            ;O
4Do2 48         DS 1            ;H
4Do3 4E         DS 1            ;N
4D04 20         DS 1            ;blenk spece
4D05 44         DS 1            ;D
4D06 4F         DS 1            ;O
4Do7 r5         DS 1            ;E
```

This program takes fewer bytes than the program we put together earlier and it is much easier to change the name to be printed. If the program seems to bomb after it writes the name on the screen, or if you want it to return to TBUG after it is through, use the B command to set a breakpoint at the end of the program (4COD for the above program). You can write any length name you want on the screen by simply loading the ASCII characters for the name into RAM at 4DOO and changing the first instruction to read LD, B,n (where n is the number of bytes in the name).

This concludes the section on the 16 bit load and jump group. Projects to work on before the next issue are:

1. Develop a program which will print the alphabet on the screen starting at location 3C10.
2. Develop a program to print every other letter of the alphabet.
3. Make a program which will put ten "A's", ten "B's" and ten "C's" on the screen in consecutive order, starting at location 3C50.

The next installment will be devoted entirely to the STACK and the CALL and RETURN instructions. ...80

realsoft tm

COMPUTER CABLEVISION inc.

(202) 337-4691

sargon
by dan and kathe spracklen
Winner of the 1978 West Coast Computer Faire Chess Tournament! Graphic display of moves. The "David" that competes against "Goliaths." Level II cessette and book with source listing. Plus commentary available.
16k, Level II $19.95 Book $14.95

microches 1.5
by peter jennings
This chess-playing program checks every move for legality, handles castling and en passant captures. You can play black or white, set up and play special board situations. 3 levels of pley.
4k, Level I or II $19.95

bridge challenger
by george duisman
You and the dummy play four person Contract Bridge against the computer. Hands can be dealt at random or according to your criterion for high card points. You can review tricks, swep sides, or replay hands.
16k, Level II $14.95

gammon challenger
by ray daly and tom throop
This backgammon-playing program requires skill and the usual luck of the dice to beat. Sharpen your play with every game against the always willing opponent.
16k, Level I or II $14.95

library 100
from the bottom shelf, inc.
One hundred programs on 5 cassettes PLUS the programming language Tiny PILOT! 25 business and finance, 15 graphic, 15 home, 15 educational and 30 games.
4k or 16k, Level II $49.95

airaid
from small systems software
Action! In this real-time shooting gallery game you shoot a ground-based missile launcher at fast small planes or slower large planes. Written in machine language with 9 levels of play. A super graphic arcase game.
4k, Level I or II $14.95

stock market
by andrew bartorillo
One to four people can play the stock market to see how much money a player can amass by "playing" 9 stocks and 1 bond. Market has a random volatility selected at start of game and allows buying on mergin using current exchange regulations. Stock splits, bull and bear markets and news items.
16k, Level II $9.95

star trek
by john christman
Four different levels of play plus a nonscrolling display make this version interesting for any Captain. Klingons move within quadrants and game is played in real-time (i.e.,centons tick by if you don't enter a move)!
16k, Level II $9.95

Insiders (tm):
TRS-80 hardware journal
with machine software
$7.50 for six monthly issues

flexI
from software associates
A machine language program which allows you to manipulate other machine language programs within memory with reading and writing tapes.
4k, Level II $12.95

renum
from software associates
Renumber any program! Merge programs! All line references are adjusted. Works for both Level II and DOS. For quality programming, this utility machine language routine program is a necssity.
16k, 32k, and 48k versions for both level II and DOS all on one tape. $14.95

tdisk
from software associates
A machine language program for disk which allows you to save machine language programs for use in DOS.
DOS $12.95

rsm 2d
from small systems software
Disk machine language monitor! Includes all the features of RSM-1S, plus reads and writes system tapes and disk sectors, has Z-80 breakpoint routine, and prints to either expansion interface or TRS232.
On disk $29.95

editor
from software associates
Disk word processing for the TRS-80! Insert or change lines, search for words or phrases, print whole or partical text, save and retrieve on disk, centers lines, right justification.
32k, DOS On cassette $39.95

COMPUTER CABLEVISION, INC.
2617 42nd St NW # 2
Washington DC 20007

Title **Price**

_____

_____

_____

_____

_____ **TOTAL**

postage and handling 1.00

_____

name _____

address _____

city & state _____

code _____ _____

☐ **Check payable to Computer Cablevision**
☐ **MASTER CHARGE** **MC bank code:**
☐ **VISA** **exp. date: /**
card number: **Signature**

DC residents only 5% tax

Barber & Beauty Shop Cash Accounting & Payroll

80-US Staff

This is the first part of a two-part article dealing with cash occounting for Borber/Beouty Shops. It is designed around a Level II 16K system, using cassette tapes. Use af o Screen or Line Printer is optionol. Although it moy nat be precisely what you can use, there ore mony techniques and ideas contained within it which may lead you ta a specialized system af yaur own. This part will caver the data accumulotion, the summory reports ond o gross cash graph. In the secand part, next issue, we will caver the payrall pragram.

Barber and Beauty Shops seem to have been left out when it comes to such things as payroll and cash accounting on micro-computers. Maybe they do not know it, but most payroll programs will not work for them. (There may be others who do not fit the norm either, but we will concentrate on Beauticians and Barbers).

Payroll programs, by and large, all base their computation on hours worked, and most employees work at some hourly rate. Others work on a salaried exempt status, but still fill out some sort of time sheet for payroll purposes.

Most beauty Shops and many Barber Shops, exist as a "collection of people doing business in one place", which is to say they are essentially self-employed. They build their own clientele and set their own price lists, based upon their experience and the demand for their services. Although in the eyes of the IRS they are still employees, they work on a commission basis, and in effect, pay the shop owner for the use of the space and facilities.

The typical shop has between three and seven employees, and it is usual to find pay arrangements where the operator will receive 50% of their gross intake up to a certain amount, more percentage above that amount, and a guaranteed minimum.

To add to the confusion of having to figure daily commissions, most shops have some sort of additional commission on retail products sold by the employee as an aid to stimulate retail sales.

The retail commission is often used to further education, as seminar fees, or as extra money at year end holiday time.

What is presented here is one way to make some sense out of this situation and get it on a computer.

THE DAILY CASH TAPE

The basis for all which follows is the collection and consolidation of the daily activity of all the operators in a given shop. Note that this information need not be entered into the computer daily, but it needs to be treated as "daily" information.

In the program which follows, lines 100 through 780 (DAILY CASH) provide the means to store daily operator service and retail sales figures on tape. It is possible to get more than three months on a C30 cassette, which includes the total transactions for each operator for each day. But let us get back to the start, and look at the source of this information.

RAW INPUT DATA

Most establishments will have some sort of cash register which will provide a "journal" tape.

By assigning each of the operators a "department" number or letter, it is possible to see from the journal tape just who did what. Two of the most popular register manufacturers, SWEDA and F-JAY, have this facility, as do many others.

If there is no cash register, or the register does not have a journal, some other means will normally be in use and can as well be used in this system.

What the DAILY CASH program does is to sum up all transactions, service and retail, for each operator on a daily basis, and store this information on a tape. By using only the beginning and ending dates, we can then go back and read in the information for any time period.

The program will also automatically keep track of service customers for each operator. This may be useful information for the shop owner or the manager in the evaluation of effectiveness of individual operators. The tape then becomes the data base for all other operations, and since some considerable time is spent entering this information, doing it only once is a fact which can be appreciated.

One of the interesting things about the DAILY CASH tape is that it need not be read from the beginning each time. It may be started in the middle of data with no ill effect, and the "fast forward" button can be used to slew up to the approximate location desired.

In the DAILY CASH program mention is made to tape (cassette) and tape (paper - from the cash register). The tape referred to in lines 420, 700, 720 and 740 is the cash register tape.

THE DAILY CASH PROGRAM

The program first asks for the date, which is entered as 011379 for Jan. 13th, 1979. The leading zero will be dropped by the computer, but it does not matter. The next three statements ask for information

concerning the amount of cash taken and change left. Later, in the summary for each day, they will provide a check to see that some of the cash did not get "lost" along the way. This is for information only, and will not be printed on the tape.

Statements 190 through 260 ask for operator number, whether the transaction is service or retail, and the amount. Lines 210, 220, 240 and 250 give you one more chance, in case you enter a number out of range.

Line 200 checks to see if the program should stay in this loop, or jump out and do the daily summeary. As you can see, entering the 30 as the operator number will cause the program to exit the loop.

Lines 270 through 400 do the job of checking to see which operator, what kind of transaction, and also count each service entry for each operator.

Since only one of these statements applies to any one entry, they all point to line 460, where, if the operator number is not 30, the program swings back to line 190, and ask for the next entry. These "IF" statements provide an accumulator for each operator, for each day.

When you enter 30 as operator number, the program jumps to line 470 and prints the days summary on the screen. You may need to change line 600 to show your particular sales tax figure. At this point, those of you with a screen printer can push the button, and have a printed record of the day. Others may want to copy the information from the screen to a ledger or whatever.

If you are making the first entry on your DAILY CASH TAPE you just get it into position now and push enter. If there are already entries on the tape, you will want to position it just past the last entries before you start. it is a good idea to mark the turns count on the tape after each session.

- THE CASH GRAF PROGRAM

This is just a little "frosting on the cake" type of thing that looks good when your business is doing well, and which you may not want to look at when it isn't. You need to re-record the program each month to update it, as the data is contained in a data statement in line 1020. Be sure when you update to delete the zero at the end and then add it again when you have updated the program. The zero is a "sentinel" which the program looks for to tell it when to stop reading data. It is possible to set up another data statement and an additional read loop to put two graphs on the screen at once. (This years totals compared to last years for example). In that case you would start the second graph at Y = 10 and give its for-next loop a step of 2 so you could tell them one from the other. Use your own imagination on that one.

The numbers in the data statement in line 1020 are for illustration only, and you can put your own real numbers there.

THE SUMMARY PROGRAM

The SUMMARY program reads the DAILY CASH tape from any date to any date, and provides totals for that period. It is handy when you need monthly or quarterly totals when filling out those many and varied forms sent to you by the City, State and Feds.

You simply run the SUMMARY program and read in the appropriate dates from the DAILY CASH tape. it gives a summary on the screen for that period. Don't be alarmed at the double solid asterisks when reading the tape. It is normal.

Also, while reading the DAILY CASH tape, the screen will display the dates between those you asked for, as they are read from the tape. It lets you know that something is happening, and that you are not hung up in some loop. It will work even when the tape is started in the middle of the data, but will ignore those dates that fall outside the ones you want.

The DAILY CASH tape collected many inputs for each operator per day. This program will collect any number of days and consolidate them into a summary with totals. It uses inputs from the DAILY CASH tape.

The program reads the tape and will check for the date in line 1570 and 1580. If the date is less than the first date you entered, the program loops back to line 1560 and enters another day, again checking the date. If the date it reads is within the range you gave it, it will print the date on the screen, and then accumulate individual operator data in lines 1600 through 1620. This will keep up till the date read in is greater than the second date you entered. Note that the data for that date is also accumulated, giving data for inclusive dates.

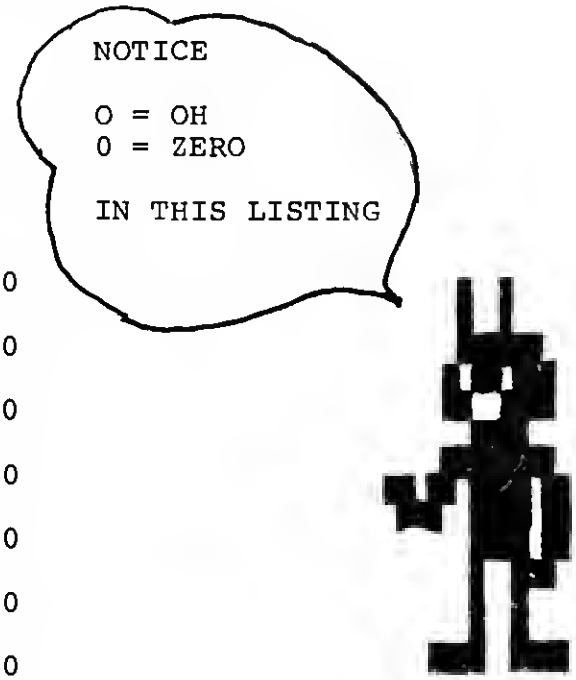
When the dates match, the program jumps to 1650 and prints the summary for the period. The .051 in line 1780 is again a sales tax figure which you may need to change.

Now you can read out summaries for monthly and quarterly tax forms, as well as any other period. In the PAYROLL program which will follow, it will be used to read in data for the pay period.

These programs, as written, will print on the screen, or on a screen printer if you have one. If you have a line printer, they will print out on it with the change of PRINT to LPRINT.

We will conclude this article in the next issue with the PAYROLL program. It is as long as these three are, and has three parts: the payroll itself, a section to create new files, and a section to audit individual operator data tapes. These programs have purposely been numbered in sequence, and the PAYROLL program starts at 2000 and goes to 3350. For those of you with enough memory, you can enter them as one large program if you prefer. The lines from 0 to 100 have been left blank, so you can add your own master index. --80

```
100 CLS:PRINT"               D A I L Y   C A S H"
110 PRINT:DIMA(12):REM (C)1979 80-NW PUBLISHING CO *
120 GOTO410
130 INPUT"TODAYS DATE IS (MMDDYY)-";A$
140 INPUT "AMOUNT TAKEN FROM SHOP TODAY"; A(7)
150 INPUT"AMOUNT OF CHANGE LEFT YESTERDAY"; A(9)
160 INPUT "AMOUNT OF CHANGE LEFT TODAY"; A(6)
170 REM * INDIVIDUAL OPERATOR INPUTS *
180 PRINT
190 INPUT"OPERATOR #         ";A
200 IFA=30GOTO470
210 IFA=0GOTO190
220 IFA>=8GOTO190
230 INPUT"SER/RET 1/2        ";B
240 IFB=0GOTO230
250 IFB>=3GOTO230
260 INPUT"AMOUNT             ";C
270 IF(A=1)*(B=1) M=M+C:E=E+1:GOTO460
280 IF(A=1)*(B=2)N=N+C:GOTO460
290 IF(A=2)*(B=1)O=O+C:F=F+1:GOTO 460
300 IF(A=2)*(B=2)P=P+C:GOTO460
310 IF(A=3)*(B=1)Q=Q+C:G=G+1:GOTO 460
320 IF(A=3)*(B=2)R=R+C:GOTO460
330 IF(A=4)*(B=1)S=S+C:H=H+1:GOTO 460
340 IF(A=4)*(B=2)T=T+C:GOTO460
350 IF(A=5)*(B=1)U=U+C:I=I+1:GOTO 460
360 IF(A=5)*(B=2)V=V+C:GOTO460
370 IF(A=6)*(B=1)W=W+C:J=J+1:GOTO 460
380 IF(A=6)*(B=2)X=X+C:GOTO460
390 IF(A=7)*(B=1)Y=Y+C:K=K+1:GOTO 460
400 IF(A=7)*(B=2)Z=Z+C:GOTO460
410 PRINT"THIS IS A DAILY CASH BREAKOUT PROGRAM"
420 PRINT"IT TAKES APART THE DAILY CASH REGISTER TAPE AND"
430 PRINT"SHOWS WHO DID WHAT FOR COMMISSION PURPOSES."
440 PRINT:PRINT"ENTER 30 AS OPERATOR # TO END PROGRAM"
450 PRINT:GOTO130
460 IFA<30 THEN 190
470 PRINT"DATE  ";A$
480 PRINT" ","SERVICE","RETAIL"," # OF CUSTOMERS"
490 PRINT "OP 1", M,N,E
500 PRINT "OP 2", O,P,F
510 PRINT "OP 3", Q,R,G
520 PRINT "OP 4", S,T,H
530 PRINT "OP 5", U,V,I
540 PRINT "OP 6", W,X,J
550 PRINT "OP 7", Y,Z,K
560 PRINT
570 A(2) = M+O+Q+S+U+W+Y
580 A(3) = N+P+R+T+V+X+Z
590 A(4) = A(2)+A(3)
600 L=A(3)*.051
610 PRINT "SERVICE","RETAIL","SALES","TOTAL SERVICE"
620 PRINT "TOTAL","TOTAL","TAX","PLUS RETAIL"
630 PRINT A(2), A(3),L,A(4)
640 A(5) = E+F+G+H+I+J+K
650 PRINT"TOTAL SERVICE CUSTOMERS FOR TODAY =";A(5)
660 A(12)=L+A(4)+A(9)-A(6)
670 IF A(12)=A(7) THEN 700
```



```
 680 IF A(12)>A(7) THEN 710
 690 IF A(12)<A(7) THEN 730
 700 PRINT"CASH AND TAPE BALANCE OK": GOTO 750
 710 A(10)=A(12)-A(7)
 720 PRINT"CASH IS SHORT OF TAPE BY";A(10):GOTO 750
 730 A(11)=A(7)-A(12)
 740 PRINT"CASH IS OVER TAPE BY";A(11):GOTO 750
 750 INPUT"POSITION DAILY CASH TAPE TO RECORD & ENTER";
 760 PRINT#-1,A$,E,F,G,H,I,J,K,M,N,O,P,Q,R,S,T,U,V,W,X,Y,Z
 770 INPUT"RECORDING COMPLETE, PRESS ENTER TO CONTINUE";
 780 RUN100
1000 CLS:REM * GROSS RECEIPTS GRAPH *
1010 GOTO1090:REM * NEXT LINE IS GROSS/MO IN HUNDREDS *
1020 DATA 50,39,32,40,56,65,72,81,68,0
1030 Y=9
1040 READ J
1050 IFJ=0THEN1400
1060 FORX=0TOJ:SET(X,Y):NEXT X
1070 Y=Y+3:GOTO1040
1080 GOTO1400
1090 PRINT@ 316,"JAN"
1100 PRINT@ 380,"FEB"
1110 PRINT@ 444,"MAR"
1120 PRINT@ 508,"APR"
1130 PRINT@ 572,"MAY"
1140 PRINT@ 636,"JUN"
1150 PRINT@ 700,"JUL"
1160 PRINT@ 764,"AUG"
1170 PRINT@ 828,"SEP"
1180 PRINT@ 892,"OCT"
1190 PRINT@ 956,"NOV"
1200 PRINT@ 1020,"DEC"
1210 PRINT@ 0,"MONTHLY GROSS RECEIPTS - HAIR FASHIONS INC"
1220 X=0
1230 FOR Y=6 TO 42
1240 SET (X,Y)
1250 NEXTY
1260 X=X+10
1270 IF X<=100 THEN 1230
1280 PRINT@ 64,"0"
1290 PRINT@ 69,"1K"
1300 PRINT@ 74,"2K"
1310 PRINT@ 79,"3K"
1320 PRINT@ 84,"4K"
1330 PRINT@ 89,"5K"
1340 PRINT@ 94,"6K"
1350 PRINT@ 99,"7K"
1360 PRINT@104,"8K"
1370 PRINT@109,"9K"
1380 PRINT@114,"10K"
1390 GOTO 1020
1400 GOTO 1400
1500 CLS:REM * CASH SUMMARY PROGRAM BY PERIOD *
1510 PRINT"                          S U M M A R Y"
1520 PRINT:PRINT" FOR WEEK, PAYPERIOD, MONTH OR QUARTER -
 WITH TOTALS FOR SERVICE/RETAIL, SALES TAX AND # CUSTOMERS"
1530 PRINT:PRINT"LOAD DAILY CASH TAPE AND SET TO READ"
```

```
1540 INPUT"STARTING DATE,I.E.(MMDDYY)";A1
1550 INPUT"ENDING DATE(INCLUSIVE)(MMDDYY)";A2
1560 INPUT#-1,A$,E,F,G,H,I,J,K,M,N,O,P,Q,R,S,T,U,V,W,X,Y,Z
1570 A3=VAL(A$)
1580 IFA3<A1GOTO1560
1590 PRINTA3;
1600 EE=EE+E:FF=FF+F:GG=GG+G:HH=HH+H:II=II+I:JJ=JJ+J:KK=KK+K
1610 MM=MM+M:NN=NN+N:OO=OO+O:PP=PP+P:QQ=QQ+Q:RR=RR+R:SS=SS+S
1620 TT=TT+T:UU=UU+U:VV=VV+V:WW=WW+W:XX=XX+X:YY=YY+Y:ZZ=ZZ+Z
1630 IFA3>=A2GOTO1650
1640 GOTO1560
1650 CLS:PRINT"SUMMARY FROM";A1;"TO";A2
1660 PRINT" ","SERVICE","RETAIL","#CUSTOMERS"
1670 PRINT"OPERATOR 1",MM,NN,EE
1680 PRINT"OPERATOR 2",OO,PP,FF
1690 PRINT"OPERATOR 3",QQ,RR,GG
1700 PRINT"OPERATOR 4",SS,TT,HH
1710 PRINT"OPERATOR 5",UU,VV,II
1720 PRINT"OPERATOR 6",WW,XX,JJ
1730 PRINT"OPERATOR 7",YY,ZZ,KK
1740 PRINT
1750 A4=MM+OO+QQ+SS+UU+WW+YY:A5=NN+PP+RR+TT+VV+XX+ZZ
1760 A6=EE+FF+GG+HH+II+JJ+KK
1770 PRINT"TOTALS",A4,A5,A6
1780 A7=A5*.051:A7=INT((A7+.5)*100)/100
1790 PRINT"TOTAL SER/RET=$";A4+A5;"        SALES TAX=$";A7
1800 STOP
```

THE MAGIC OF LEO CHRISTOPHERSON

M. Schmidt
Editor

Back in November 78 we hinted that Leo Christopherson, the author of Android Nim, was quietly working on something new and different. This is a report on some of his efforts...

The magic of Leo Christopherson started with ANDROID NIM, back in September 1978. It was featured on the cover of 80-NW (now 80-US) in November. We called it "exquisite", and referred to it as "setting the standard" for all animated graphics on the '80, trying not to be overly hyped up about it.

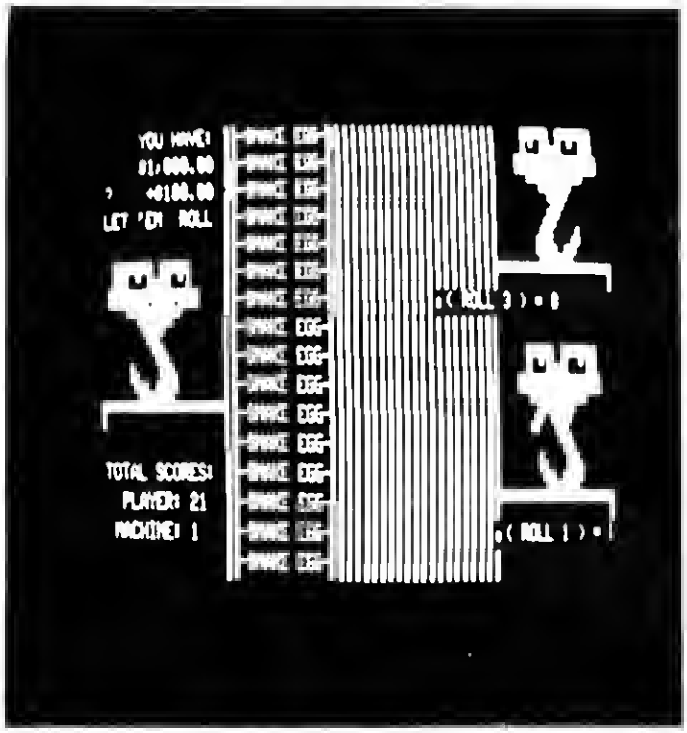
It has turned out to be all that and more. If the number of "bootleg" copies are any indication of popularity, it certainly has it.

Leo didn't rest easy after ANDROID though, his next step being to add more animation and sound to the Androids. They now open their mouths and utter cries as they close in for the kill!

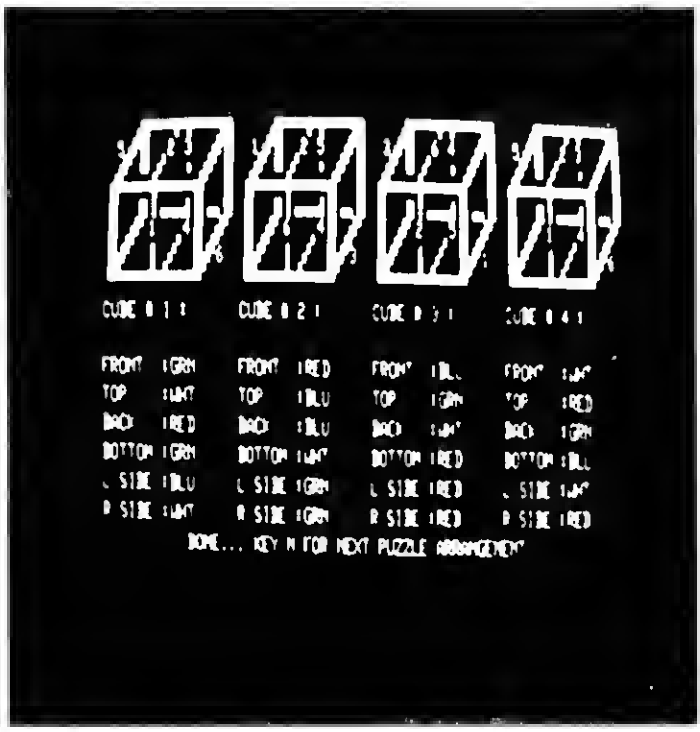
But how about space? The original Android was a tight squeeze in 16K; how could you add more animation plus sound? After the oils of many midnights burned low, Leo found a way to pack string elements into about one third less space than they previously occupied. Then he introduced a unique method of inserting machine language routines into the basic program. Next, he packed the machine language routines, same way as he did the strings.

Just one problem now, and that was how to get the programs to run in Level II and also in DOS. (The USR command is different).

The answer was to find some other command to use as a vector to the machine language routine, one that was common to both L2 and DOS. LPRINT was found to work for ANDROID and was used for that program, but now, after running the program, the line printer will not respond unless you reset the system.



SNAKE EGG



CUBES

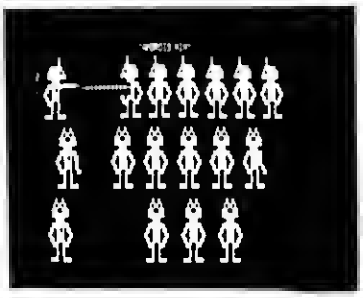
With much help from T Rosenbaum, it was determined that an unused command would be better, and sure enough, there is one in Level II - NAME. So now all the programs Leo writes use NAME as a vector to machine language routines. It can be placed anywhere in the program, and does not leave anything upset when you have finished running.

One of the by-products of all this is that the listings look very much like a "bad load". The string packing leaves lines of program looking like endless Level II command words (A$="USINGUSINGUSINGDEF DEFDEFDEFKILLKILL") for example. Oh yes, they take up room on the screen, but in the memory, each of those command words is actually one byte, so that instead of CHR$(191) you now have just one byte to represent those nine in memory!

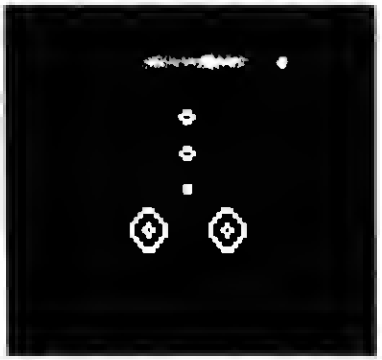
In listing on the screen it can also cause the display to scroll, and it will drive a printer listing bonkers!

Now it was not the intention to do this, it just happened as a result of saving memory space. The way it is done is to build a subroutine with high line numbers, run the strings through it, and later delete the subroutine. Sort of like locking the door and throwing away the key. Perhaps in a future article, we can get Leo to give us the nitty-gritty of it all.

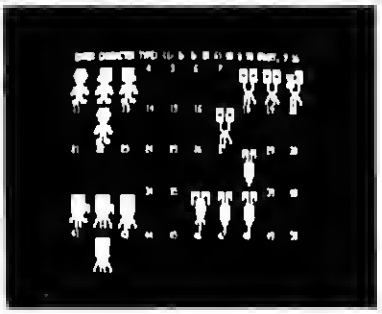
Now that he has found his new method, he has gone and written more. SNAKE EGGS has three snakes playing a game against you which is difficult to describe. Loosly based on the game of "21" (times two), they lay eggs which knock down 21 lines. A 21 gives you a "Snake Egg", and over 21 gets a "Scrambled Egg" for you (and you lose). There are the refusals to do what they are told to do, as in Android, but now they pass remarks back to each other, and the same refusal will not always invoke the same remark. The sound on this one is really an integral part of the game, and there is a "Call to the Post" at the beginning, a Chopin funeral dirge when you lose and a Sousa march when you win!



ANDROID NIM
with SOUND



LIFE at 100
gen/minute!



THE BATTLE
of LIFE

Then, there is his new game of LIFE, called LIFETWO because he has packed two games into one 16K program. The first option gives a normal (?) game of life, with the screen displaying 100 generations per minute! The second part has four different kinds of Leo's "creatures" playing the game of life to see which life form will survive. It too, has a varied range of sound effects.

Another creation by Leo is the game called "CUBES". This one will give you the solution to "Instant Insanity", you simply tell the computer the orientation of the colored (or numbered) blocks and then watch the computer go out of it's mind trying to arrange them so that no two colors or numbers are adjacent. It lists the solution (if there is one) at the end. The systematic trial and error is clearly shown on the screen as the computer tries every possible combination. Sweet revenge if you have ever spent a whole afternoon trying to do the same thing. Incidentally, "Instant Insanity" is a registered trade mark of Parker Brothers, a Division of General Mills Fun Group, Inc.

What is next for Leo? He won't talk until the product is finished, but he is working up more - something about an "Alien Encounter" - but we will have to wait and see.

How do you get sound? Simple: just plug the connection which now goes to the AUX of your cassette recorder into any audio amplifier. RS has one called the Realistic 200 mw Solid State Speaker Amplifier (Cat. #277-1008) which sells for around $10.

Actually, any audio amplifier will work, and the higher the "FI", the better. Have fun! ..80

SCIENTIFIC CALCULATIONS ON LEVEL I

T.R. Dettmann, Associate Editor

Who says you can't do some pretty heavy things with Level I machines? Here is an example of just what can be done in level I, using a little thought and ingenuity. Although this may be a "far out" idea for most (pardon the pun), it again shows the diverse application possible with even minimum hardware.

Important calculations on the computer are not done because of the availability of a high level language or because a particular language offers powerful ways of expressing mathematical concepts. Instead, the calculations are done because they need to be done. We often forget that many of the really important calculations have been done in science, mathematics, business etc, with minimal equipment.

The first computers could do no more than add and shift data, and yet they were used for some of the most important calculations of their day. Even modern computers translate programs from a high level language down to the level of the machine before the steps are executed.

Your computer does not know how to multiply, it is the program in ROM which accomplishes the multiplication of two numbers by addition and shifting. Programming languages such as Level II Basic just make it more convenient for the programmer by taking the responsibility from him for the calculations involved on the computer's level. Because of this fact, even a limited language such as Level I can do significant calculations.

To show you how this is done, the program listing offers a program to do a significant calculation in Astrophysics in Level I. The problem which it addresses is to find the limits of spiral structure in the galaxy. In simpler terms, we want to find the closest point to the center of the galaxy that we could have spiral structure as well as the farthest point. The theory for the calculation involves some considerable playing with complicated mathematics, but it can be reduced to a few fairly simple formulas which, when plotted as a function of the radius of the galaxy, will give the desired limits visually.

Most everyone knows that we presently believe the galaxy is a gigantic spiral, revolving around a nucleus with billions of stars. The problem of how the arms got to be there has plagued astronomers for years. Even more of a problem is the fact that the arms should wrap around the nucleus and eventually disappear if they are made of any real material.

For years, one scheme after another was invented by astronomers to explain why the arms of the galaxy still exist. One theory explained them by saying that they were periodically regenerated by explosions in the center of the galaxy. There is evidence that there may have been such explosions, but it seems rather unlikely that they happen regularly enough to cause the arms.

Another theory would have us believe the spiral arms were actually magnetic bottles which kept their shape as well as keeping stars and gas inside. Unfortunately, this predicted that we would see things that never actually happened.

In the early 1960's, a brilliant mathematician, named Lin, suggested that we were looking at the problem incorrectly. He said that the only reason we saw more stars in the arms was that they were in a "gravity well" around the nucleus of the galaxy, and so they stayed there longer than outside the "well" because it was hard to get out, just the same kind of problem as you might have when you climb down into a valley: it is easy to get down but climbing out the other side is tough. Working from this theory, astronomers have predicted that we would see star formation in the arms, and we do see young stars. They also predicted we should see two arm structures, and in fact these seem to predominate.

In fact, the theory explains many observations that previously were unconnected. As with any theory, there are holes and limits due to idealizing the system, but some form of the present theory will certainly be a part of any extensions in the future.

The present calculations, to determine the limits of spiral structure is based on this theory. It requires that we compare the rotational speed of stars going around the galaxy as we go further and further from the nucleus. These speeds can be calculated from observational data which is normally expressed as a "rotation curve". This simply gives the angular velocity of stars around the nucleus as we

get further from the nucleus. We also calculate from theory the speed of the "pattern" of the spiral, how fast the gravitational well is circling the galaxy. When we compare the two, the inner and outer limits to the spiral structure can be found by finding the value of the rotation speed where the pattern and the matter rotate at the same speed.

The limits will then occur at plus and minus one half a number called the Epicyclic Frequency from there. The theory gives all the formulas in the notation of calculus, but with a little work and some approximations, the formulas can be reduced to a form that is only algebra.

For example, in calculus a quantity called a derivative measures the rate of change of some number. If we plotted a graph of the number, this would be the same as measuring the slope of the graph. Since we can express the function in this way, we can program a derivative by calculating the slope. Each function of this sort can be broken down to simpler and simpler concepts until we get it to a programmable form.

The program uses approximations like those above to calculate these inner and outer limits to spiral structures, (called inner and outer LINDBLAD resonances). It does it in a little over 2K RAM on a Level I machine and draws a graph of the result on the screen at the end. There are some interesting points in the program which can be used in more than just this one particular program.

Subroutine 30010 is the square root subroutine from the Level I manual. It provides the program with the ability to use the square root in the calculation. Even more important when dealing with a large complicated program, are routines like these beginning at lines 1000 and 1100. These routines test out subroutines of the master program independently of the program itself. They allow error checking the program in modules instead of hoping the result of the whole program is right. This can really be a help when you make changes to the subroutines, since these error checking modules can check their results on data for which you know the result.

Subroutine 10000 is provided to automatically hold the screen display every 10 lines of output and wait for operator action to continue. In this way, when you do not have a printer, you can copy the information from the screen to use elsewhere.

Subroutine 22000 plots a graph of the results on the screen so that you can verify the results and estimate the answer visually for comparison. The "print at" statements on lines 22120 to 22140 put labels on the display. A picture of such a screen display can make a good hard copy for later use. Even a Polaroid picture can get enough

```
 5 REM (C)1978 80-NW PUBLISHING CO
 10 REM CALCULATE EPICYCLIC FREQUENCY AND LOCATE
 20 REM LINDBLAD RESONANCES FOR THE GALAXY
 30 REM VERSION 2.0 MODIFIED TO COMPUTE WITH MORE NEARLY
 40 REM CORRECT ROTATION CURVE, 11/26/78
 95 E=1:F=30:G=1
 98 GOSUB 10005
 100 FOR I= E TO F
 105 R=I*G
 110 GOSUB 21000
 120 K=(4*O*O)*(1+R/(2*O))*D)
 122 H=0:IF K<0 THEN H=1
 123 IF K<0 THEN K=-K
 125 X=K:GOSUB 30010:K=Y
 126 A=O-K/2:B=O+K/2
 127 A(I)=R:A(50+I)=O:A(100+I)=A:A(150+I)=B
 128 IF R =15 THEN C=O
 130 T=I/10 - INT(I/10)
 135 IF H=0 THEN PRINT R,O,A,B
 136 IF H=1 THEN PRINT R,O,A,B; TAB(2),"*"
 140 IF T=0 THEN GOSUB 10000
 160 NEXT I
 170 GOSUB 22000
 999 END
 1000 REM TEST ROTATION CURVE
 1005 CLS
 1010 FOR R= 1 TO 40
 1020 GOSUB 20000
 1030 X=3*R
 1035 Y=47-47*V/250
 1036 IF V<0 THEN Y=47
 1040 SET(X,Y)
 1050 NEXT R
 1060 GOTO 1060
 1100 REM TEST SQUARE ROOT FUNCTION
 1105 CLS
 1110 FOR X = -1 TO 25
 1120 GOSUB 30010
 1130 PRINT X,Y
 1140 FOR I= 1 TO 50: NEXT I
 1150 NEXT X
```

NOTICE

O = OH
0 = ZERO

IN THIS LISTING

```
9999 END
10000 INPUT"PRESS ENTER TO CONTINUE";A$
10005 CLS
10010 PRINT"GALACTIC ROTATION CURVE"
10020 PRINT
10030 PRINT"R","OMEGA","OMEGA-K/2","OMEGA+K/2"
10040 PRINT
10050 RETURN
20000 REM ROTATION CURVE FOR GALAXY
20010 IF R<=3 THEN V=(200/3)*R
20020 IF (R>3)*(R<=9) THEN V=(50/6)*(R-3)+200
20025 IF R>9 THEN 20060
20035 IF V<0 THEN V=0
20040 D=-Q/R-V/(R*F)
20050 RETURN
20060 V=250-(50/16)*(R-9)
20070 GOTO 20035
20100 V=100*R
20110 D=100/R-V/(R*R)
20120 RETURN
20200 V=200
20210 D=-V/(R*R)
20220 RETURN
21000 REM COMPUTES ANGULAR VELOCITY
21010 GOSUB 20000
21020 O=V/R
21030 RETURN
22000 CLS:REM PLOT ROTATION CURVE DATA ON SCREEN
22009 REM M=VERTICAL SCALE FACTOR
22010 M=50
22020 FOR I=1 TO 47:Y=I:SET(0,Y):SET(1,I):NEXT I
22030 FOR I=1 TO127:X=1:SET(X,47):NEXT I
22040 FOR I=1 TO 30
22045 X=4*A(I)
22050 Y=47-47*A(50+I)/M:IFY>47 THEN Y=47
22055 IF Y<0 THEN Y=0
22060 SET (X,Y)
22070 Y=47-47*A(100+I)/M:IF Y>47 THEN Y=47
22075 IF Y<0 THEN Y=0
22080 SET (X,Y)
22090 Y=47-47*A(150+I)/M:IF Y>47 THEN Y=47
22093 IF Y<0 THEN Y=0
22095 SET (X,Y)
22100 Y=47-47*C/M:SET(X,Y)
22110 NEXT I
22120 PRINT AT 0,"200";:PRINT AT 448,"100";:PRINT AT960,"0";
22130  PRINTAT1000,"20";:PRINTAT980,"10";:PRINTAT1020,"30";
22140 PRINTAT280,"LINDBLAD RESONANCES";
22150 GOTO 22150
30010 REM SQUARE ROOT
30020 IF X=0 THEN Y=0: RETURN
30030 IF X>0 THEN 30060
30040 PRINT "NEGATIVE NUMBER":Y=0:RETURN
30060 Y=X*.5:Z=0
30070 W=(X/Y-Y)*.5
30080 IF(W=0)+(W=Z) THEN RETURN
30090 Y=Y+W:Z=W:GOTO 30070
```

resolution to be used in a preliminary report.

The program as written is not a hard one to write or use. It depends only on understanding your problem well enough to be able to reduce it to the simplest possible terms. Computer scientists have been doing just that for years when they write the complex programs and make them execute on a micro-processor which can only understand addition and data movement. (If you don't believe it, look at the commands in the Z80 language, multiplication has to be programmed, it is not a "canned" function).

Level I users do not have to limit themselves to playing games or balancing their checkbook, waiting for the power of a Level II system. Any problem which can be expressed clearly and simply can be programmed, even in Level I ...80

bluffit

A GAME OF BLUFF AND COUNTER BLUFF

Roy Groth, Graham, Wa.

This Level II game was developed by Roy (who is 15 years of age) by writing the program at home where he has no system, and then checking it out on the computer at his school. That seems like doing it the hard way, but we like his efforts and think you will too.

Bluffit (also known as Issac's Bluffer), is an exciting game of bluff and counter bluff for two players, one of which is your computer.

The rules are simple, but the stratedy can become mind-boggling with paradoxical intricacies. Do you really trust your computer? - or?

These are the rules: The deck consists of eleven cards, all the aces, kings and queens (except the queen of spades).

Five cards are dealt to each player, and the extra card is placed face down between the players. The object of the game is to correctly guess the identity of the odd card, or trick your opponent into guessing incorrectly.

To get information about the odd card, the players alternate turns. In a turn a player can do one of two things: He can guess at the odd card, by making it known that he is guessing, stating what he thinks the odd card is, and turning the odd card over. If the player guessed correctly it is a win. If player guesses wrong it is a loss. A guess always ends the game.

The player has another option which is to call out the name of a card that has not already been named. Then, if the opponent has the named card it must be put on the table face up. If the opponent does not have the card a "no" answer must be given. The turn then goes to the other player.

That is all there is to the rules. However, the strategy involved is the interesting part of the game.

First, let us say it is your turn, and you call for a card that is not in your hand. If your opponent answers "no" you know that you have just asked for the odd card.

Then, all you have to do is to wait until your next turn, and guess the odd card, winning the game.

But--your opponent knows all this too, so when he answers "no", since his turn comes before you get a chance to guess, he can guess the odd card first and win. Right? -- Well, maybe.

Note that nothing in the rules says you must ask for a card that is not in your hand. Asking for a card in your hand seems useless though. You won't learn anything, since your opponent will answer "no". But if he doesn't realize you were doing this, he would guess wrong on the next turn, and you may win the game. This, of course, is known as the bluff.

Bluffing all the time does not help you much though, try it and see. The best strategy is to mix your bluffs and straight calls, to keep your opponent unsure.

You should either hope to trick him with one of your bluffs, or have him think you bluffed when you really did not. Keep in mind the opponent will be doing the same thing to you!

To play the computer load the program and run it. Your five cards will appear in the upper left corner of the screen. In the upper right of the screen there will be five rows of X's. These are the backs of the computer's cards.

In the center of the screen there will be another row of X's, this being the back of the odd card.

The turn to start is random. If it is the computer's turn, one of two things can happen. Usually, the name of a card will be displayed a few rows below the odd card. This means that the computer is asking for that card.

You must answer yes if you have it and no if you do not. Do not try to cheat. The computer will catch you at it, and it does not tolerate cheating!

If you answer yes when the computer asks for a card an asterisk will appear by that card. This is to signify that the card is now face up.

If the computer is guessing, it will print "I guess" (name of card). The odd card will then turn over, and the outcome of game decided. The total score will then be displayed. Press enter for the next game.

When it is your turn, a question mark will appear a few rows beneath the odd card. Enter the name of the card you wish to ask for. If what you typed disappears when you press enter, you either mis-typed, or asked for a card which has already been turned up. The computer will then respond "yes", and turn the card over, or "no".

If you wish to guess, type "guess" instead of a card. Then enter the name of the card you think it is. The odd card will then turn over, the game will be scored, and the computer will wait for another game.

That is all there is to Bluffit. And, to answer you when you have lost twelve games to six: no, the computer is not looking at your cards, even tho it seems like it must be.

Note: Do not use Line Feed (down arrow) in line 40
Type line 40 as a continuous line..

```
1 REM (C)1979 80-NW PUBLISHING * BLUFFIT BY ROY GROTH *
3 CLEAR150:DEFINTA-Z
4 RANDOM
5 DIM IB(5,5),HB(5,5),CD(11),CD$(11),HC(5),IC(5)
10 HB=0:IS=0:HS=0:FD=RND(2)
20 RESTORE:DATA 33,50,50,56,58,26,33,37,40,43,19,26,29,31,33,
16,21,23,25,27,12,17,19,21,22:FORI=1TO5:FORJ=1TO5:READIB(I,J):
NEXTJ,I:
30 DATA 50,50,41,38,33,33,33,28,25,22,38,32,27,24,21,32,30,26,
23,21,32,29,25,23,21:FORI=1TO5:FORJ=1TO5:READHB(I,J):NEXTJ,I
40 DATA "ACE OF SPADES","ACE OF DIAMONDS","ACE OF HEARTS","ACE
 OF CLUBS","KING OF SPADES","KING OF DIAMONDS","KING OF HEARTS"
,"KING OF CLUBS","QUEEN OF DIAMONDS","QUEEN OF HEARTS","QUEEN
OF CLUBS":FORI=1TO11:READCD$(I):NEXTI
47 RANDOM
50 FORI=1TO11:CD(I)=0:NEXTI:FD=3-FD:HN=5:IN=5:FG=0
60 FORI=1TO5
70 CD=RND(11):IFCD(CD)=1GOTO70
80 CD(CD)=1:HC(I)=CD                                    NOTICE
90 CD=RND(11):IFCD(CD)=1GOTO90
100 CD(CD)=1:IC(I)=CD:NEXTI                             O = OH
110 FORI=1TO11:IFCD(I)=1NEXTI                           0 = ZERO
120 TC=I:CD(I)=1:GOSUB520
130 IFFD=2GOTO270                                       IN THIS LISTING
140 IFRND(100)<=IB(HN,IN)GOTO220
150 GU=RND(11):IFCD(GU)=0GOTO150
160 FORI=1TO5:IFIC(I)=GUGOTO150ELSENEXTI
170 CD(GU)=0:PRINT@600,CD$(GU);"       ":INPUTAN$:IFAN$="YES"
GOTO200
180 IFAN$<>"NO"GOTO170
190 FG=1:GOTO270
200 FORI=1TO5:IFHC(I)=GUTHENHC(I)=-GU
210 NEXTI:GOSUB520:HN=HN-1:GOTO270
220 GU=RND(11):IFCD(GU)=0GOTO220
230 FORI=1TO5:IFIC(I)=GUGOTO240ELSENEXTI
235 GOTO220
240 CD(GU)=0:PRINT@600,CD$(GU);"       ":INPUTAN$:IFAN$="YES"
GOTO510
250 IFAN$<>"NO"GOTO240
260 IN=IN-1
270 PRINT@600,"                    ":PRINT:PRINT@600,"";:INPUT
AN$:IFAN$="GUESS"GOTO460
280 FORCD=1TO11:IFCD$(CD)=AN$GOTO300ELSENEXTCD
290 GOTO270
300 IFCD(CD)=0GOTO270
310 CD(CD)=0:FORI=1TO5:IFIC(I)=CDGOTO500ELSENEXTI
320 RANDOM:PRINT@640,"NO":FORI=1TO400:NEXTI:IFRND(100)+HB>=HB
(HN,IN)GOTO390
330 IFFG=0GU=CD
340 PRINT@600,"I GUESS ";CD$(GU):FORI=1TO400:NEXTI:PRINT@336,
CD$(TC);"         "
350 IFTC=GUGOTO380
```

```
355 IFFG=1GOTO510
360 PRINT@640,STRING$(128," "):PRINT@640,"I LOSE":HS=HS+1:HB=
HB+10
370 PRINT"MY SCORE:";IS,"YOUR SCORE:";HS:INPUTAN$:GOTO50
380 PRINT@640,STRING$(128," "):PRINT@640,"I WIN":IS=IS+1:HB=
HB-5:GOTO370
390 HN=HN-1
400 IFFG=1GOTO340
410 IFHN=0GOTO450
420 IFIN<>0GOTO140
430 FORGU=1TO11:IFCD(GU)<>0GOTO340ELSENEXTGU
450 FORI=1TO5:CD(ABS(IC(I)))=0:NEXTI:GOTO430
460 INPUT"YOUR GUESS IS";AN$:FORCD=1TO11:IFCD$(CD)=AN$GOTO480
ELSENEXTCD
470 GOTO270
480 FORI=1TO400:NEXTI:PRINT@336,CD$(TC);"       "
490 IFTC=CDGOTO360ELSEGOTO380
500 PRINT@640,"YES      ":FORJ=1TO400:NEXTJ:IC(I)=-CD:GOSUB520:
IN=IN-1:GOTO400
510 CLS:PRINT"YOU CHEATER!!! I'M NOT PLAYING WITH YOU ANY MORE"
:END
520 CLS:PRINT@0,"";:FORI=1TO5:PRINTCD$(ABS(HC(I)));:IFHC(I)<0
PRINT" *";
530 PRINT TAB(32);:IFIC(I)<0GOTO550
540 PRINTSTRING$(17,"X"):GOTO560
550 PRINTCD$(ABS(IC(I)))
560 NEXTI:PRINT@336,STRING$(17,"X"):RETURN
```

SOUND

SOUND

SOUND

SOUND

SOUND

SOUND

SOUND

SOUND

ANIMATED GRAPHICS AT IT'S BEST — WITH SOUND!

ANDROID NIM
SNAKE EGG
LIFETWO

OFFERED TO YOU BY 80-US

SOFTWARE - FOR YOUR TRS-80

SEE PAGE 45 FOR COMPLETE LISTING OF PROGRAMS AND ORDER BLANK

HANGUPS

Reference Bowling (again!), W O Eden, Eugene, Oregon askes if it is possible to get a straight ball out of the game. If you really want to become a 300 bowler, remove lines 240, 242, 245 and 247. This will give you a perfectly straight ball every time.

Louise Frankenberg, Pasadena, Maryland says that since she upgraded to 16K RAM, the Print CHR$(23) command blanks out everything on the screen until she hits CLS or Clear. Does anyone else have this probelm? - the CHR$(23) command is supposed to shift the screen display into 32 character mode. The CLS and Clear will normally get you back to the normal 64 character mode from a CHR$(23) command. We suggest you take it back and have whoever put in the 16K upgrade take another look at it. All we can say for sure is that it is not right.

Truman Krumholz, Springfield, Missouri tells us that the BIO-2 Program (Nov-Dec 78 80-NW) has an error, in that it will not accept February 29th as a birthday. he is right! And the correction he sent along fixes the problem, so here it is - in line 280 add to the end of the line: ELSE 310. Add to end of line 300: ELSE320, and to the end of line 310 add: ELSE RETURN Thanks Truman.

From Bob Weinberg, Mentor, Ohio, comes the following: "One bug which has reared its ugly head for me involves the 'In-Memory Information System'. I have found that any data tapes created under software control of this package in Level I cannot be used later in Level II. You can imagine my dismay after hours of inputting inventory items, quantities, reorder levels, descriptions etc., soon after I bought Level I in Feb 78; only to find after waiting ages for a Level II kit that I'd have to start all over. Who out there knows enough to be able to write additional programs which can utilize the data tapes created; or even expand on them?"

That is a tough one Bob, we have heard of, but never ever seen, a conversion tape for the In-Memory stuff. Perhaps one of our readers has an answer?

Here is one from Philip Litchfield, New Canaan, Connecticut which represents the questions from many other Amateur Radio Operators: "Many of your readers will be, as I am, Amateur Radio Operators as well as TRS-80 owners. Therein lies the problem - interfacing the TRS-80 with Amateur Radio Teletype Machines, particularly the ASCII ones that are (or will soon become permissible, we hope) used by HAMs - the 33ASR and the 35RO using a 20 milliamp loop. A discussion of the RS-232 in your column would be greatly appreciated. I guess I have missed everything written on it. It would solve the problems here in my 'Shack', and a whole lot of others might find teletype gear a lot cheaper (though much larger) than printers."

We will have a review of the RS-232 in the May-Jun issue, and are working up an article specifically devoted to the TRS-80 and RTTY, nothing definite on that yet, but it is in the works. Being an ex-HAM (W7JJW) myself, I can see some exciting possibilities in this area. (Mike)

That's it for this time, thanks again for the input.

Tom & Mike

UNCLASSIFIED ADS

TRS-80 YELLOW PAGE on hardware & software. Send $1.00 plus 2 SASE for 2 issues. Vendors send $20.00 & product description. 96 Dothan St. Arlington, MA 02174

TRS-80 L2 - ORIGINAL SOFTWARE. Music, Magic & of course Games. Keyboard: Play melodies on your TRS-80. Card Trick: Computer guesses card(s) chosen by up to 3 people. 5-Speed Racer: Real time action. Arithmetic Drill: Giant numerals & right to left entry; + - * / $5.00 each or all 4 for $15.00. SASE for complete list to ADVANCED GRAPHICS 920 W. Romneya #6. Anaheim, CA 92801

LOST-COST PRINT! Asm'd-Tstd hardwre-sftwre to use Mod 15 TTY as TRS-80 Line Prtr-$34.95-Also Cheap Selectric out soon-LTI 219 E Jacob-Midwest City, OK 73110

WE HAVE AN RS 232 serial interface, plug compatable w/TRS-80 Expansion interface w/software & instr. Precision Electronics, 3411 S 90th St. Tacoma, WA 98409

OWNERS OR USERS OF TRS-80 L2 or DOS systems interested in doing community/humanitarian service drop a postcard with name, address & type equip to Mike Freeman, 946 N Alder St. Tacoma, WA 98406

YOUR AD COULD BE HERE FOR ONLY $2.50 A COLUMN INCH

A NOTE ON DISK BASIC

The location of DOS in RAM has been a puzzle, but a little bit of research turned up the following memory map for the DOS system:

0000-3000	Level II BASIC ROM
37DE-37EC	I/O Addresses
3800-3BFF	Keyboard Memory
3C00-3FFF	Video Display Memory Map
4000-41FF	BASIC Vectors
4200-51FF	TRSDOS
5200-6FFF	Disk BASIC, DOS Utilities (when loaded) or User Memory
7000-FFFF	User Memory not used by DOS

If you are working with machine language programs that you want to be compatible with Disk BASIC or DOS Utilities, you must start at memory locations 7000 Hex or greater to avoid being overwritten.

If you are running machine language programs in Disk BASIC, there are a lot of things in memory that you can use. Here is a random collection of a few useful items:

1. Store the address of an 8 byte buffer in HL, then CALL 446DH will put the time into the buffer, and CALL 4470H will put the date in the buffer

2. memory address 4318H is the address of a 64 byte buffer that contains the last DOS command entered

3. 40B1 is the least significant part of the address of the last unprotected memory location in memory after a MEMORY SIZE protect.

4. 57FO is the location of a 61 characterASCII format copyright by MICROSOFT

ANOTHER NOTE ON BASIC PROGRAMMING

Working with graphics all the time, it is usually helpful to be able to draw the display as it will show on the screen before you try to program it. Both the Level I and Level II Basic Manuals give a video display worksheet for the screen (Level I, page 106 - Level II on page E/1). It is hard to use these pages though, since one use makes it hard to use it again, and repeated use will destroy it.

One answer is to print up copies of these pages and use one per project. This is nice for documentation, and if that is your aim, you might as well do it, but a much simpler way is to either make a single copy of the page or remove the page and encase it in an 8 x 10" piece of clear adhesive plastic. (It is available from most office supply stores or college book stores). You can easily write on this with a grease pencil, but these are messy and never really clean up properly.

The best choice for working out graphics on such a sheet is a transparancy pen such as the "Vis-a-Vis", by Sanford. It is available for 69 cents at office supply stores. They come in a variety of colors, will stay put almost indefinately, but will wipe off easily with a damp rag.

Use it to work out graphics displays, tables, options lists, etc. on your acetate covered video display worksheet. The cost is negligible, but the ease that it gives to working out the screen displays is worth its weight in gold.

NOTE ON DISK BASIC
(or, some neat places to go in memory)

Here's a collection of interesting places to jump to in memory using the SYSTEM command in level II or DISK BASIC or DEBUG in DOS (or any other monitor usch as TBUG, RSM, etc.):

0000	reboots DOS in command mode (same as powerup)
402D	enter DOS without reinitializing
5200	takes you back to the entry point for DISK BASIC if its in memory (gives you the 'Number of Files' message

...80

LEVEL II POKE GRAPHICS

George Blank, Leechburg , PA.

Here is a handy reference which you can use for the development of grahic displays. You may want to make a copy of this page and place it between two sheets of plastic, so that you can prop it up next to your computer for handy reference. While you are at it, make a list of the error codes (or the ASCII codes), and put it back-to-back with this page before you encase them in plastic. That way your reference card will be more useful!

LEVEL II GRAPHICS USING POKE FUNCTIONS

The "PRINT AT" locations on your screen can be divided into six graphics elements. See fig 1, which represents one print at location on your screen. Normally, one character would occupy this space with a margin around it. In graphics though, turning on all six of the elements in all the locations on the screen will "white out" the screen.

Here is a handy little chart that will greatly simplify using POKE graphics. To use it, look at the Block Map and determine the letters of the blocks you wish to set. Then look for the code that corresponds to those letters. For example, if you wanted to light the 3 blocks in a vertical row on the left side, the Block Map tells you that you want A,C,E. Looking at the chart, you can determine that you need to POKE 149 in the desired location on the screen. Remember that 128 is a space, and you can reset any block by POKE{Memory},128.

This programming trick will also make it easy to use the Video Display Worksheet to lay out your POKE graphics. Set a variable equal to 15360, and then simply add the PRINT AT location to that variable to establish your memory location. This allows you to use the PRINT AT location directly, and greatly simplifies later revision and debugging. This sample program uses this technique to light the middle two blocks {C & D} of PRINT location 544 in the middle of the screen.

```
100 A=15360
110 POKE A+544, 140
```

space	128	a f	161
a	129	b c	134
a b	131	b c d	142
a b c	135	b c d e	158
a b c d	143	b c d e f	190
a b c d e	159	b c d f	174
a b c d e f	191	b c e	150
a b c d f	175	b c e f	182
a b c e	151	b c f	166
a b c e f	183	b d	138
a b c f	167	b d e	154
a b d	139	b d e f	186
a b d e	155	b d f	170
a b d e f	187	b e	146
a b d f	171	b e f	178
a b e	147	b f	162
a b e f	179	c	132
a b f	163	c d	140
a c	133	c d e	156
a c d	141	c d e f	188
a c d e	157	c d f	172
a c d e f	189	c e	148
a c d f	173	c e f	180
a c e	149	c f	164
a c e f	181	d	136
a c f	165	d e	152
a d	137	d e f	184
a d e	153	d f	168
a d e f	185	e	144
a d f	169	e f	176
a e	145	f	160
a e f	177		

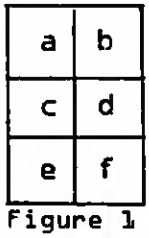


Figure 1

80-US SOFTWARE
PRESENTS
ANIMATION plus SOUND!

1. NEW! ANDROID NIM with SOUND! The TRS-80's first animated graphics winner now has more animation and SOUND - by Leo Christopherson Level II, 16K $14.95

2. NEW! LIFETWO with SOUND! Conways game of life at an astaunding 100 generations per minute! PLUS Leo C's animated "talking" creatures playing the "BATTLE OF LIFE" in ane 16K, Level II program $14.95

3. NEW! SNAKE EGG! with SOUND This version of "21", with talking snakes, will keep you enthralled for some timel by Lea Christapherson $14.95

DEALER INQUIRIES INVITED

1. NEW! CUBES! Another Lea Christopherson special! Gives the salution to "Instant Insanity" (r) or numbered blocks. Let your camputer go crazy trying to figure how to arrange the blocks on your screen. Level II, 16K $9.95

2. SPECIAL! The original ANDROID NIM. While they last! The first TRS-80 animated graphics game only $8.00

3. TEXT80 A simple Basic Text Editor with no complicated sub-cammands you need to learn. On a formatted disk (or customer supplied DOS disk) $12.00

4. Available after 1 May 79: TEXT/SET a unique basic text editor/print formatter program for 32K and 48K TRS-80 Disk Systems. Has complete control in editing and printing and is "operator oriented". With comprehensive user manual. On formatted disk (or customer supplied DOS) Price ta be announced.

All 80-US Software is sent prepaid First Class

SUBSCRIPTION to the **80-US JOURNAL** *is the ONLY way to keep up on what's happening. You can pay more (or less), but you will not find the VALUE anywhere, except in the JOURNAL! We guarantee it. Use the order form to the right (or a reasonable facsimile - we aren't picky). If you are already a subscriber - do something nice, tell a friend about the* **JOURNAL.**

VISA

master charge

BACK ISSUES OF 80-US

Issue No. 1 - not ovoilable
☐ **Issue No. 2 - $3.00**
☐ **Issue No. 3 - $3.00**

80-US SOFTWARE

Send to:
80-US Journal
P.O. Box 7112
Tacoma, WA 98407

Please order by the name of the program **AMOUNT**

NAME .....

.....

.....

.....

WASHINGTON RESIDENTS ADD 5.1% SALES TAX
MASTER AND VISA CARDS GLADLY ACCEPTED

NAME _____

ADDRESS _____

CITY, STATE, ZIP _____

80-US SUBSCRIPTION ORDER FORM

MAIL TO:

80-US Journal
P.O. Box 7112
Tacoma, WA. 98407

$16.00 a year

MASTER AND VISA CHARGE GLADLY ACCEPTED

Starting with the _____ issue

NAME _____

ADDRESS _____

CITY, STATE, ZIP _____

Level II TRS-80 + The LIBRARY 100

You shouldn't have one without the other!!

Finance: Present Value of a Future Sum—Simple Interest for Days—Future Value of a Present Sum—Amortization Schedule—Interest Rate: Compound Interest—Interest Rate: Installment Loan—Days Between Dates—Term of an Installment Loan—Present Value of a Series of Payments—Real Estate Capital Investment—Nominal and Effective Interest Rates—Internal Rate of Return—Future Value of Regular Deposits—Regular Deposits for Future Value—Depreciation Amount: Rate; Salvage Value: Schedule—Bond Present Value—Bond Yield to Maturity—Sale—Cost—Margin—Day of the Week—Moving Ad.
Education: Multiplication & Division—Add—Subtract—Fraction & Decimal—States & Capitals—States & Order of Entry—States & Abbreviation—Inventors & Inventions—World Capitals & Countries—Urban Areas & Population—Authors & Books—Presidents & Order—States & Largest City—Basenum.
Graphics: Left Right—Random Ad—Graphic—Blocks—Fireside—Snow—Step Ad—Step Ad 2—Launch—Ratrace—War Game—Weird—Herring—Blinker—Snoopy.
Home: Message Board—Expense Account—Nutrition—Mileage—Remember—Phone Codes—Night Check Off—Drunkometer—Perpetual Calendar—Babysitter—Calculator—Bartender—Christmas List—Vacation Check Off—Conversion.
Games: Speedy—Odd One—R. Roulette—Star Blazer—Search—Spyship—Tiger Shark—Jumble 2—Sting Ray—Stars—Sketch—Flipper—Scissors—Horse—Doomsday—Craps—Jumble 1—Mem. Quiz Letters—Mem. Quiz Numbers—Wheel of Fortune—Decision—Unjumble—Fifteen—Towers—Life—Star Trek—Race Track—Count—Roachrace—Gypsy.

The Library 100 is the best buy in software history! We do have one problem, though. Some people cannot believe that we can give you so much for only $49.50. We suggest that if you don't believe it, you ask your friends who have purchased this software package or go to your local computer store and ask to see it.

The Library 100 is a collection of 100 original, quality programs, plus a new, easy programming language for your TRS-80, Tiny PILOT (TPILOT).

It took us 37 pages in our manual just to say a little about each program, so we cannot describe them all here. We did list them. How many can you use?

You've read the announcements in Kilobaud and Creative Computing. It is here and it is real. A value in excess of $500.00 for only $49.50 plus $2.00 postage and handling (plus 4% for Georgia residents).

Watch for future software announcements. Hardware? Calculator? Business systems? Write and tell us what you want next.

The Bottom Shelf - Library 100

NOW AVAILABLE FROM 80-US

Save $2 postage & handling - only $49.50 postpaid

P.O. Box 7112, Tacoma, WA 98407 *Wash. Residents add 5.1% tax*

MasterCharge & VISA accepted - give full information -

COMPUTER SYSTEMS CENTER

226½ West Foothill Blvd.
Claremont, CA. 91711

714/621-5059

TRS-80 TM Mods

1 Upper/Lower Case
installed with switch between "jibberish" and standard Level II. With software for Level II to turn "jibberish" into properly displayed data.

2 Level I/II
installed in same machine switch selected.

3 Clock Rate
switch selected 1.77MHZ (standard) or 2.66 MHZ (fast) cut your processing time by 1/3.

TRS-80 TM Components

4 Serial Interface
installed in keyboard. With software driver for Level II

5 16 K RAM
installed & tested

6 16KRAM
chips only (you install)

7 Floppy Disc
Pertec FD-400 with cable, enclosure & power supply. We will test on your expansion interface

8 Integrat Data Printer
pressure feed platten, upper/lower case, programmable character size, 50 c.p.s. continuous impact printer.

9 Centronics Up-1
as item 8 but using silvered paper (see in R.S. catalog).

10 12" B&W TV
modified for TRS-80. Switchable as TV or video monitor. Can be used with B&W camera.

Package units properly. We are not responsible for shipping damage.

TRS80 System L2 16K
Complete - $849.00

Item	Price	You send
1	49.00	keyboard
2	29.00	keyboard
3	29.00	keyboard
4	49.00	keyboard
5	119.00	keyboard or exp. intf.
6	99.00	nothing
7	399.00	exp. optional
8	849.00	exp. optional
9	449.00	exp. optional
10	149.00	nothing

MICRO MANAGEMENT SYSTEMS

SPECIAL PRICE ON TRS80 EQUIPMENT

MICRO-COMPUTER SPECIALIST
LARRY OWENS

RADIO SHACK DEALER - MINI MALL
DOWNTOWN SHOPPING CENTER
CAIRO, GA. 31728

912-377-7120

GAMES

Basic Games For TRS-80 L 2 or DOS, 4 or 16K

We have over 25 different L 2 games on two cassettes or two disks.

301	16 L2 4K games	$20.00
401	10 L2 16K games	$20.00
501	12 L3 32K games	$25.00
601	8 L3 48K games	$25.00

THESE ARE ALL NEW GAMES
And For Less Than $2.00 Per GAME YOU Can't Beat The Prices.

SOFTWARE, UNLTD.
3411 South 90th St.
Tacoma, Wa. 98409

80-US
PO Box 7112
Tacoma, WA 98407

Bulk Rate
US Postage
PAID
Permit No.774
Tacoma, WA